

ADRIANA LUIZA DO PRADO

***MODELAGEM MATEMÁTICA DO FLUXO DE
TRÁFEGO VEICULAR***

CURITIBA

2010

ADRIANA LUIZA DO PRADO

***MODELAGEM MATEMÁTICA DO FLUXO DE
TRÁFEGO VEICULAR***

Tese aprovada para obtenção do grau de Doutor em Ciências, na área de concentração de Programação Matemática, do programa de Pós-Graduação em Métodos Numéricos em Engenharia, promovido pelos Setores de Ciências Exatas e de Tecnologia da Universidade Federal do Paraná.

Orientadora Profa. Dra. Liliana Madalena Gramani

Co-orientador Prof. Dr. Eloy Kaviski

CURITIBA

2010

Dedicatória

A Deus,

A minha mãe Maria pelo estímulo e ao meu filho Luciano
pela paciência.

Agradecimentos

À professora Liliana Madalena Gramani, pela compreensão e dedicação com que orientou este trabalho, tornando possível a realização do mesmo.

Ao professor Eloy Kaviski que contribui com seus conhecimentos, os quais foram essenciais para este trabalho.

À minha mãe D. Maria e ao meu filho Luciano pelos longos dias de espera e privações .

Aos colegas DMAT-UFPR e do PPGMNE-CESEC, que contribuíram com seus conhecimentos, os quais foram essenciais para minha formação .

Ao prof. Cristovão Vicente Scapulatempo Fernandes, pelas suas sugestões.

E por fim a todas as pessoas que, direta ou indiretamente, contribuíram para a realização deste trabalho e dessa grande vitória pessoal o meu mais profundo agradecimento.

Epígrafe

Ser feliz
não é um sonho
mas uma questão
de decisão.
(Dercy Gonçalves)

Sumário

| | |
|--|------------|
| Lista de Figuras | ix |
| Lista de Tabelas | x |
| Resumo | xi |
| Abstract | xii |
| 1 Introdução | 1 |
| 1.1 Escalas de representação | 1 |
| 1.2 A relação entre o fluxo e o comportamento do motorista como partícula ativa . | 2 |
| 1.3 A influência das condições iniciais na geração da função de distribuição | 2 |
| 1.4 Justificativa e objetivos | 3 |
| 1.4.1 Justificativa | 6 |
| 1.5 Contribuições | 7 |
| 1.6 Organização do trabalho | 7 |
| 2 Revisão dos modelos de tráfego | 9 |
| 2.1 Descrição microscópica | 9 |
| 2.1.1 Modelo seguindo o líder | 10 |
| 2.2 Descrição macroscópica | 11 |
| 2.2.1 Modelo de Lighthill-Whitham | 14 |
| 2.2.2 A equação de Burgers | 17 |
| 2.2.3 Modelos de Prigogine e Phillips | 18 |
| 2.2.4 Modelos de Whitham et al. | 19 |
| 2.3 Descrição cinética | 20 |
| 2.3.1 Modelo de Prigogine tipo Boltzmann | 21 |
| 2.3.2 Modelo de Paveri Fontana (MPF) | 23 |
| 2.3.2.1 Modelo para Velocidades Previstas Especificadas para MPF . | 25 |

| | | |
|----------|--|-----------|
| 3 | Método das características e método das diferenças finitas | 28 |
| 3.1 | Equações macroscópicas do tráfego | 28 |
| 3.2 | Método das características (mc) | 30 |
| 3.2.1 | O processo geral do método das características | 34 |
| 3.3 | Simulação numérica do método das características | 39 |
| 3.4 | Aspectos conclusivos para o método das características | 41 |
| 3.5 | Método de diferenças finitas explícito | 43 |
| 3.6 | Simulação numérica do método de diferenças finitas explícito (mdf) | 44 |
| 3.6.1 | Para uma pista circular | 44 |
| 3.6.2 | Comparação entre o método das características com o método das diferenças finitas | 47 |
| 3.6.3 | Para uma pista única com uma entrada lateral | 48 |
| 3.7 | Aspectos conclusivos para o método das diferenças finitas explícito | 50 |
| 4 | Modelo da velocidade discreta | 52 |
| 4.1 | Caso espacialmente homogêneo | 54 |
| 4.2 | Representação do sistema veículo-motorista através da partícula ativa | 56 |
| 4.2.1 | Fundamentação teórica | 57 |
| 4.2.2 | Perfil dos motoristas | 58 |
| 4.2.3 | Introdução ao método dos volumes finitos | 62 |
| 4.2.4 | Resultados da simulação do motorista | 63 |
| 4.2.5 | Conclusões do perfil do motorista | 65 |
| 4.3 | Análise dos efeitos de condições iniciais aleatórias nos resultados das simulações | 66 |
| 4.3.1 | Números Pseudo-Randômicos | 67 |
| 4.3.1.1 | Gerador congruente multiplicativo (MC) | 68 |
| 4.3.1.2 | Gerador por deslocamento de registro (SR) | 70 |
| 4.3.1.3 | Gerador com defasagens de Fibonacci (LF) | 71 |
| 4.3.1.4 | Geradores combinados | 72 |
| 4.3.1.5 | Gerador RANLUX | 73 |
| 4.3.2 | Algoritmo de Metropolis | 75 |
| 4.3.3 | As equações de evolução para funções randômicas | 76 |
| 4.3.4 | Método de Runge-Kutta de 4ª Ordem | 79 |
| 4.3.5 | Análise das distribuições de probabilidade na geração das condições iniciais | 80 |
| 4.3.6 | Conclusões da análise das funções aleatórias | 82 |

| | | |
|----------|---|-----------|
| 5 | Conclusões e Recomendações | 84 |
| 6 | Nomenclaturas e normas das rodovias em geral | 86 |
| 6.1 | Código de Trânsito Brasileiro | 86 |
| 6.2 | Estradas de Rodagem | 87 |
| | Referências | 93 |
| | Anexo A - Informações em Anexos | 97 |
| | Apêndice A - Programas em Anexos | 99 |

Lista de Figuras

| | | |
|------|--|----|
| 2.1 | Fig. 1 do artigo Pipes(1953) | 10 |
| 2.2 | Diagrama fundamental da densidade | 13 |
| 3.3 | Discretização da malha. | 35 |
| 3.4 | Gráfico no plano xt das características, sobre as quais encontra-se a solução. . . | 35 |
| 3.5 | Valores da condição inicial | 40 |
| 3.6 | Variação de ρ pelo método das características | 42 |
| 3.7 | Variação da velocidade v e da densidade ρ para sete curvas características . . . | 43 |
| 3.8 | Evolução da densidade veicular ρ (veic./km) pelo mdm no apêndice A, anexo (2) | 46 |
| 3.9 | Evolução da velocidade média v (km) pelo mdm no apêndice A, anexo (2) . . . | 46 |
| 3.10 | Erro Relativo vezes 10^{-4} da densidade veicular (ρ) por métodos mc e mdm x Frequência acumulada | |
| 3.11 | Erro Relativo vezes 10^{-4} da velocidade média (v) por mc e mdm x Frequência acumulada | 48 |
| 3.12 | Evolução da densidade de carros entrando | 49 |
| 3.13 | Velocidade em função da posição na estrada no apêndice A, anexo (3) | 49 |
| 3.14 | Densidade em função da posição na estrada no apêndice A, anexo (3) | 50 |
| 4.15 | Probabilidade P versus densidade ρ (caso 1) | 59 |
| 4.16 | Probabilidade P versus densidade ρ (caso 2) | 59 |
| 4.17 | Probabilidade P versus densidade ρ (caso 3) | 60 |
| 4.18 | Probabilidade P versus densidade ρ (caso 4) | 61 |
| 4.19 | Probabilidade P versus densidade ρ (caso 5) | 61 |
| 4.20 | Resultado da velocidade para simular o caso 1 | 65 |
| 4.21 | Resultado da densidade para simular o caso 1 | 65 |
| 4.22 | Velocidade média v versus a densidade ρ apêndice A, anexo (5). | 81 |
| 4.23 | Fluxo médio q versus a densidade ρ , apêndice A, anexo (5). | 82 |

Lista de Tabelas

| | | |
|-----|--|----|
| 3.1 | Valores de $v(x, t)$ para a posição x e para nove valores de t | 41 |
| 4.2 | Relação de casos analisados | 58 |
| 4.3 | Resultados para o caso 4. | 60 |
| 4.4 | Algoritmo do Método do Volumes Finitos | 63 |
| 4.5 | Resultado da simulação para o caso 1. | 64 |
| 4.6 | Resultado da simulação para o caso 4. | 64 |

Resumo

O estudo científico do fluxo de tráfego começou nos anos trinta com a aplicação da teoria das probabilidades para a descrição do tráfego rodoviário e modelos de estudo que relacionou o volume, velocidade e investigação de desempenho do tráfego nos cruzamentos. Os fenômenos de fluxo de tráfego, como quase todos os sistemas do mundo real, pode ser observado e representado em diferentes escalas. Entretanto artigos sobre este tema não têm mostrado resposta satisfatória quanto à escala adequada ou iterações características do sistema veículo-motorista. O principal objectivo deste estudo é descrever a cinética do fluxo de veículos e examinaram os efeitos das condições iniciais na geração de funções aleatórias, bem como a influência do comportamento do motorista sobre o fluxo de tráfego. Usa-se programas de simulação numérica em Fortran com as fórmulas da mecânica de fluidos, método das características e das diferenças finitas.

Palavras-chave: Teoria de tráfego veicular, Escala Cinética, Método das Diferenças Finitas, Simulação.

Abstract

The scientific study of traffic flow started in the thirties with the application of probability theory to describe the traffic and study models that related the volume, speed and performance investigation of traffic at intersections. The phenomena of traffic flow, as almost all real-world systems, can be observed and represented at different scales. However articles on this topic have not shown satisfactory response as to the appropriate scale or iterations characteristics of vehicle system pilot. The main objective of this study is to describe the kinetics of the flow of vehicles and examined the effects of initial conditions on the generation of random functions, as well as the influence of driver behavior on traffic flow. It uses numerical simulation programs in Fortran with the formulas of fluid mechanics, method of characteristics and finite difference.

Key-words: Theory of Traffic Flow, Kinetic Scale, Finite Diference Method, Simulation.

1 Introdução

Neste trabalho apresenta-se uma descrição cinética para o fluxo de tráfego veicular. Para isto serão apresentados as escalas de representação: microscópica, macroscópica e mesoscópica ou cinética e o porquê da escolha cinética. A escala cinética é um estado intermediário onde o sistema é identificado pela posição e pela velocidade dos veículos, mas não se referindo a cada veículo e sim à uma distribuição de probabilidade adequada. As variáveis que conduziram a pesquisar este assunto estão relacionadas ao fato de que na medida que cresce o fluxo de tráfego aumenta significativamente a poluição do ar, engarrafamentos diários e as externalidades causando impacto nas atividades socio-econômicas. Seus aspectos matemáticos, físicos e ambientais nos instigaram a modelar e simular situações do fluxo de tráfego. Eventualmente poderá servir para analisar o sistema que existe e mostrar alternativas.

Estudou-se na escala cinética a relação entre o fluxo e o comportamento do motorista como partícula estatística ativa, a influência das condições iniciais na geração da funções de distribuição para uma pista única e a modelagem para o caso espacialmente homogêneo (quando não há variação com a posição x).

1.1 Escalas de representação

A modelagem do fenômeno do fluxo de tráfego pode ser desenvolvida por diferentes representações de escalas: microscópica, macroscópica e mesoscópica ou cinética. A escala microscópica refere-se à identificação individual dos veículos. Neste caso, a posição e a velocidade de cada veículo definem o estado do sistema como variáveis dependentes do tempo, ou seja, sua modelagem é baseada na Mecânica Newtoniana. Na escala macroscópica o tráfego é associado

ao escoamento de um fluido e o comportamento individual dos veículos é ignorado. A modelagem macroscópica é baseada na hidrodinâmica utilizando-se as equações de conservação e modelos fenomenológicos. Para a modelagem cinética o estado do sistema é identificado pela posição e pela velocidade dos veículos. No entanto esta identificação, não se refere a cada veículo, mas à distribuição de probabilidade adequada ao longo do estado microscópico considerada como uma variável aleatória. Os modelos cinéticos descrevem a evolução da função de distribuição por equações não-lineares integro-diferenciais com uma estrutura semelhante a equação de Boltzmann¹ que descreve a dinâmica de um gás ideal. A equação a seguir destaca a estrutura básica do modelo:

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} + \frac{F}{m} \frac{\partial f}{\partial v} = \frac{\partial f}{\partial t} \Big|_{colisao} \quad (1.1)$$

sendo que f representa a função de distribuição de uma partícula; F representa a força; m representa a massa da partícula; t representa o tempo e u representa a velocidade das partículas.

1.2 A relação entre o fluxo e o comportamento do motorista como partícula ativa

Pesquisa-se o comportamento do motorista como uma partícula ativa. A partícula ativa representa a qualidade do motorista (neste caso é a condição emocional do motorista: calmo, agitado e agressivo) em um sistema veículo-motorista. A densidade de probabilidade total da função de distribuição é representada como função da variável discreta que assume-se ser constante. Propõe-se um método para determinar a distribuição do perfil dos motoristas.

1.3 A influência das condições iniciais na geração da função de distribuição

Apresenta-se uma análise dos efeitos das condições iniciais estabelecidas por diferentes métodos de geração de números pseudo-aleatórios uniformes, sobre os resultados obtidos. Para

¹Anexo A: anexo 1

tanto foram investigados 18 algoritmos para geração de números pseudo-aleatórios uniformes para gerar a função de distribuição inicial f que aparece na equação 1.1. A função de distribuição de uma partícula é a função que fornece o número de partículas por unidade de volume num espaço de fase².

1.4 Justificativa e objetivos

As teorias do fluxo de tráfego buscam descrever de uma maneira matemática as interações entre veículos, motoristas e a infra-estrutura. A infra-estrutura consiste no sistema da estrada e em todos os seus elementos operacionais, incluindo dispositivos de controle, semáforos e sinais de trânsito. Estas teorias são indispensáveis em todos os modelos de tráfego e ferramentas para à análise de operações nas ruas e estradas. O estudo científico do fluxo de tráfego teve seu início na década de trinta com a aplicação da teoria de probabilidade à descrição do tráfego de estrada e com os estudos de modelos que relacionavam o volume, a velocidade e a investigação do desempenho do tráfego nos cruzamentos. Com o aumento significativo do uso dos automóveis e a expansão do sistema viário surgiu o estudo de características do tráfego, envolvendo modelos descritos por equações diferenciais e íntegro-diferenciais, ou seja, modelos aplicáveis na área da teoria cinética³.

Os modelos matemáticos aplicados na teoria cinética foram desenvolvidos, depois dos trabalhos pioneiros de Prigogine e Herman (1971), para o modelo de fluxo de tráfego veicular sobre estradas e redes de estradas. As idéias de Prigogine motivaram a pesquisa neste campo por vários autores entre outros Paveri Fontana (1975) que trabalharam para resolver as equações macroscópicas do fluxo de tráfego derivadas das equações do tipo-Boltzmann levando-se em conta o comportamento individual do veículo usando modelos específicos da velocidade desejada.

Para Nelson (1995) o fluxo de tráfego em uma única pista (sem ultrapassagem) com todos os veículos com a mesma velocidade desejada é considerado, olhando o desenvolvimento de

²Anexo A: anexo 2

³Anexo A: anexo 3

uma equação cinética ⁴ na qual as interações da aceleração são tratadas da mesma maneira que tradicionalmente se tratam as interações de desaceleração. Tal equação cinética é desenvolvida baseada em um modelo específico de correspondência manifestando a principal função de distribuição de veículos, que fornece o número de veículos por unidade de volume num espaço de fase, em termos de uma única função de distribuição de veículos e de um modelo específico da mecânica que descreve as circunstâncias e a forma na qual os motoristas mudam sua velocidade em resposta a sua situação com relação ao veículo imediatamente à frente deles. O modelo mecânico emprega uma extensão de forma limitada da hipótese de caos que classicamente tem sido empregada para o termo que representa as interações da desaceleração (a hipótese de caos veicular generalizado é baseado na hipótese do caos molecular que diz respeito à correlação entre duas moléculas entretanto nada afirma sobre a forma da função de distribuição (Huang, 1987)). As distribuições que representam soluções (local) de equilíbrio da equação cinética com interações de veículos individuais são indicados para compor uma família de distribuições bimodal, com picos na velocidade nula e na velocidade desejada. Interpretações e as consequências de tais equilíbrios são discutidos do ponto de vista da teoria do fluxo de tráfego, especialmente no que diz respeito ao modelo correspondente ao fluxo de tráfego (relação do fluxo-densidade) e o tráfego de para-e-anda.

Adicionalmente Klar, Kühne e Wegener (1996) estudaram os diferentes tipos de modelos matemáticos usados para a simulação de tráfego de veículos. O artigo inclui modelos baseados em equações diferenciais ordinárias, equações de dinâmica de fluidos e em equações do tipo Boltzmann. As ligações entre os diferentes tipos de modelos são mencionadas baseada em modelos cinéticos e em métodos de simulação para os modelos.

Mais geral e não menos completo, Helbing (2001) faz uma revisão geral dos modelos de tráfego considerando dados empíricos, fatos e observações. Também desenvolve as principais abordagens sobre pedestres e tráfego de veículos.

Para Darbha e Rajagopal (2002) o fluxo de tráfego é descrito através de uma abordagem da continuidade de um fluido compressível⁵, uma abordagem estatística através da teoria cinética

⁴Anexo A: anexo 4

⁵Anexo A: anexo 5

dos gases ou de modelos de autômatos celulares. Embora tais sistemas sejam grandes coleções (conjuntos ilimitados), eles não sejam grandes o suficiente para que sejam tratados como um continuum⁶.

Como alternativa, podemos desenvolver uma abordagem de sistemas dinâmicos discretos que é particularmente adequado para descrever a dinâmica dos sistemas de grande porte como o tráfego.

Bellomo, Coscia e Delitala (2002) referem-se principalmente à modelagem por equações de dinâmica de fluidos e modelagem cinética, mostrando alguns aspectos metodológicos da modelagem matemática e para a interpretação dos resultados experimentais.

Kerner (2004) fornece uma interpretação física detalhada dos fenômenos do tráfego que focalizam os vários eventos observados que devem ser corretamente descritos por modelos matemáticos.

A estrutura geral do ponto de vista físico é fundamental para se obter a modelagem matemática adequada. O número de partículas ativas no sistema é constante no tempo, entretanto, não são homogeneamente distribuídas no espaço. Os veículos devem ser considerados como partículas ativas cujas propriedades mecânicas devem ser integradas modelando o papel do motorista, que difere de veículo para veículo. Ou seja, os motoristas podem ser experientes ou não, agressivos ou não, etc. Desta forma, as propriedades dos veículos diferem de caso para caso, conseqüentemente suas características específicas não podem ser consideradas constantes no sistema. Conseqüentemente, o raciocínio anterior sugere incluir uma descrição apropriada para modelar as características específicas do sistema motorista-veículo da estrada.

Bellomo e Coscia (2005) analisaram criticamente os modelos da hidrodinâmica de primeira ordem⁷ do fluxo de tráfego de veículos obtidos pelo fechamento⁸ através de uma relação matemática fenomenológica⁹ entre a velocidade média local e o perfil da densidade local. Comenta-se que o veículo não pode ser considerado simplesmente como um sistema mecânico, que deve ser pen-

⁶Anexo A: anexo 6

⁷Anexo A: anexo 7

⁸Anexo A: anexo 8

⁹Anexo A: anexo 9

sado como um sistema veículo-motorista no qual a habilidade do motorista de se adaptar não deve ser desprezada. Vários modelos determinísticos e estocásticos são descritos e analisados.

Os modelos determinísticos são modelos de simulação que não contém nenhuma variável aleatória, ou seja, para um conjunto conhecido de dados de entrada ter-se-á um único conjunto de resultados de saída. Os modelos estocásticos possuem uma ou mais variáveis aleatórias como entrada, que levam a saídas aleatórias. É utilizado quando pelo menos uma das características operacionais é dada por uma função de probabilidade.

1.4.1 Justificativa

Os fenômenos do fluxo de tráfego, como quase todos os sistemas do mundo real, podem ser observados e representados por escalas diferentes. Na literatura, a maioria dos trabalhos baseiam-se em modelos que descrevem a interação entre veículos. A modelagem dos fenômenos do tráfego por métodos da teoria cinética e os artigos citados anteriormente, ainda não apresentam respostas satisfatórias quanto a:

1. A seleção da escala apropriada para a descrição do sistema: microscópica (individual), macroscópica (médias globais) e mesoscópica (mistura da micro e da macro);
2. A avaliação das variáveis que identificam as características específicas do sistema do motorista-veículo, e portanto as iterações entre estas características e a dinâmica.

Esses são os pontos que serão abordados. A interpretação física do fluxo de tráfego veicular e a qualidade descritiva de fenômenos específicos é uma referência essencial para o modelo matemático. Certamente, requer-se que os modelos possuam a habilidade de descrever o comportamento qualitativo dos fenômenos específicos do fluxo de tráfego, quando as comparações quantitativas sejam difíceis, ou mesmo impossíveis, devido a casualidade e a irregularidade do fluxo de tráfego.

Os veículos devem ser modelados como partículas ativas, considerando que suas propriedades mecânicas precisam ser integradas junto com o papel do motorista, que varia de veículo

para veículo. Em outras palavras, o motorista pode ser experiente ou não, tímido ou agressivo, etc. Inclusive as propriedades do veículo diferem de caso para caso, conseqüentemente, as suas características específicas não podem ser considerados parâmetros uniformes do sistema. Desta forma usa-se variáveis aleatórias.

1.5 Contribuições

Iniciou-se o trabalho com as equações macroscópicas do fluxo de tráfego derivadas das equações reduzidas de Pavari-Fontana para modelos com valores especificados da velocidade prevista para uma pista através do método das características (Prado et al., 2006) e pelo método das diferenças finitas explícito considerando um problema de condições de contorno periódicas e outro problema de fluxo em um trecho de estrada com entrada lateral (Prado et al., 2007).

Com o desenvolvimento buscou-se outras escalas de representação do fluxo de tráfego, optou-se pela escala cinética. Usou-se programas de simulação numérica em Fortran com as fórmulas da mecânica de fluidos, método das características, método das diferenças finitas explícito e método dos volumes finitos.

Examinou-se a influência das condições iniciais na geração de funções aleatórias para uma pista única, a relação entre o fluxo e o comportamento do motorista como partícula ativa e modelagem para o caso espacialmente homogêneo (quando não há variação com a posição x).

1.6 Organização do trabalho

No Capítulo 2 apresenta-se uma revisão dos modelos de tráfego encontrados na literatura e as principais ideias de cada um destes modelos. O Capítulo 3 aborda um dos primeiros trabalhos realizado nesta pesquisa utilizando a descrição macroscópica e as equações reduzidas de Pavari-Fontana com simulações numéricas utilizando o método das características e de diferenças finitas explícito.

No capítulo 4 mostra-se o modelo da velocidade discreta, as duas linhas que existem de velocidade discreta e introduz-se o modelo espacialmente homogêneo trabalhados nas seções

4.2 e 4.3.

Por fim as conclusões obtidas neste trabalho são mostradas no Capítulo 5.

2 *Revisão dos modelos de tráfego*

Os fenômenos do fluxo de tráfego, como quase todos os sistemas do mundo real, podem ser observados e representados por escalas diferentes, entre elas: microscópica, macroscópica e mesoscópica ou cinética. Cada escala trabalha com equações e relações específicas. Os modelos microscópicos podem contribuir para descrever matematicamente a interação entre veículos em modelos cinéticos, enquanto os modelos macroscópicos relacionam às equações empregadas em métodos assintóticos apropriados às equações cinéticas.

2.1 *Descrição microscópica*

A descrição microscópica refere-se à identificação individual dos veículos. Neste caso, a posição e a velocidade de cada veículo definem o estado do sistema como variáveis do tempo. Os modelos matemáticos consistem na determinação de uma equação diferencial para a dinâmica de cada veículo, baseada na mecânica newtoniana, sob a ação dos veículos circundantes.

Os elementos desta escala são:

- $x(t)$ é a distância em função do tempo;
- $v(t) = \frac{dx}{dt}$ é a velocidade em função do tempo e
- $a(t) = \frac{dv}{dt} = \frac{d^2x}{dt^2}$ é a aceleração em função do tempo.

Para condições iniciais denotadas por t_0 , x_0 e v_0 tem-se a equação:

$$x(t) = x_0 + \int_{t_0}^t v_0 dt + \int_{t_0}^t \int_{t_0}^t a(t) dt dt. \quad (2.1)$$

Considerando n veículos na estrada temos:

$$a_i(t) = \frac{dv_i}{dt} = \frac{d^2x_i}{dt^2} \text{ com } i = 1, \dots, n. \quad (2.2)$$

A solução do sistema de equações diferenciais ordinárias fornece a descrição do escoamento do sistema.

2.1.1 Modelo seguindo o líder

O primeiro modelo foi proposto pelo físico Pipes (1953), que analisou a dinâmica de uma linha de tráfego, composta de n veículos. Postulou que os movimentos dos vários veículos são controlados por uma lei ideal de separação baseada no Código de Trânsito da Califórnia. A lei considera que cada veículo deve manter aproximadamente cinco metros, a cada dezesseis quilômetros por hora, de distância do veículo anterior. Esta distância garante pelo menos um segundo de tempo de reação para o motorista caso alguma coisa aconteça a sua frente.

Também verificou-se que quando a luz em um cruzamento fica verde, toda a linha de veículos controlados por ele não se move como uma unidade, mas como uma onda de partida. Observações têm revelado o fato de que a velocidade desta onda tem o valor aproximadamente constante de 48 quilômetros por hora ao longo da fila de veículos. As equações da dinâmica que regem a linha de tráfego podem ser obtidas, exigindo que cada veículo obedeça a lei acima.

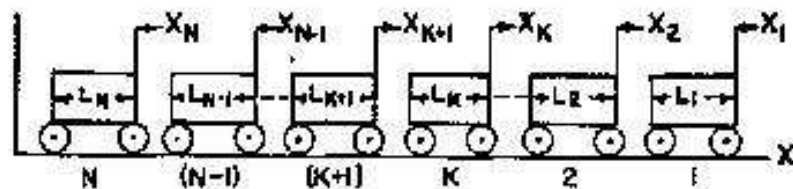


Figura 2.1: Fig. 1 do artigo Pipes(1953)

Com relação a Figura 2.1, utilizou-se das seguinte notações: x_k como coordenada do k -ésimo veículo (em pés), $b + Tv_k$ a distância legal (manter 5 metros a cada 16 quilômetros

por hora) e L_k o comprimento de um carro (aproximadamente 5 metros). Quando esta lei de trânsito é obedecida ele observou que dois veículos sucessivos da linha, de coordenadas x_k e x_{k+1} , devem satisfazer o seguinte conjunto de equações:

$$x_k = x_{k+1} + (b + Tv_{k+1}) + L_k, \quad k = 1, 2, 3, 4, \dots, (n-1). \quad (2.3)$$

Se estas equações são diferenciadas em função do tempo, o resultado é,

$$\dot{x}_k + Tv_{k+1} = v_k, \quad k = 1, 2, 3, 4, \dots, (n-1). \quad (2.4)$$

Estas equações (2.4) podem ser reescritas em termos das velocidades dos veículos, sob a forma

$$Tv_{k+1} + v_{k+1} = v_k, \quad k = 1, 2, 3, 4, \dots, (n-1). \quad (2.5)$$

As equações (2.5) são chamadas equações dinâmicas do sistema de veículos, este conjunto de equações foi fundamental no estudo do movimento da linha de veículos sob a postulada lei de trânsito da Califórnia e foi utilizado para estudar casos especiais de significado prático.

As implicações destas investigações são de que certos parâmetros fundamentais que envolvem características do motorista podem ser descobertos e incorporados no design dos veículos e estradas, para melhorar o fluxo de tráfego e evitar congestionamentos.

2.2 Descrição macroscópica

Nesta seção descreve-se alguns modelos macroscópicos, entre eles: modelo de Lighthill-Whitham (o mais antigo), equação de Burgers (melhoramento do modelo de Lighthill-Whitham), modelo de Prigogine e Phillips (outra alternativa) e o modelo de Whitham et al. (generalização do modelo de Lighthill-Whitham).

Fisicamente o comportamento individual dos veículos é ignorado e o fluxo de tráfego pode

ser associado a um escoamento de um fluido.

A descrição macroscópica refere-se a representação do estado do sistema por uma média de quantidades (quantidades macroscópicas), ou seja, a densidade e o momento linear, consideradas como variáveis dependentes do tempo e do espaço.

A modelagem é baseada nas equações de conservação da massa, do momento e da energia como nos modelos da hidrodinâmica e modelos fenomenológicos.

O fluxo de tráfego é modelado como um fluido compressível formado por veículos e descrito por uma estrutura macroscópica considerando que o sistema pode acumular um número máximo de veículos e que os veículos circulam a uma dada velocidade média máxima.

Os elementos desta escala são:

- $\rho(t, x)$ é a densidade de veículos, sendo nula quando não se tem veículos no sistema e aumenta conforme o número de veículos aumenta.
- v_{max} é a velocidade máxima média dos veículos e atinge o seu maior valor no caso de fluxo livre.
- $q(t, x)$ é o fluxo de veículos, sendo zero quando não se tem veículos no sistema ou quando a densidade de veículos é máxima. A densidade de veículos situa-se no intervalo $[0, \rho_{max}]$ e a densidade máxima ρ_{max} é alcançada quando os veículos não podem mais se mover (congestionamento/engarrafamento).

A velocidade máxima poderá ser superada por um veículo veloz e isolado movendo-se com velocidade limite v_l . Neste caso as experiências mostram que a velocidade limite v_l é afetada por vários fatores, entre eles, a qualidade da pista e dos veículos. Em particular, a velocidade limite pode ser definida como $v_l = (1 + \mu)v_{max}$, $\mu > 0$, sendo μ um parâmetro fenomenológico.

A relação entre densidade, velocidade e fluxo é dada pela equação:

$$q = f(\rho) = \rho v, \quad (2.6)$$

conhecida como equação fundamental e o diagrama decorrente desta equação: é o diagrama fundamental, ilustrado na Figura 2.2.

Na Figura 2.2 a região anterior a ρ_{crit} é denominada de região de fluxo livre (ou região estável) e a região posterior a ρ_{crit} é denominada de região de fluxo congestionado (ou região instável). Ainda observa-se que para densidades diferentes, por exemplo, ρ_A e ρ_B , tem-se o mesmo fluxo, $q_A = q_B$.

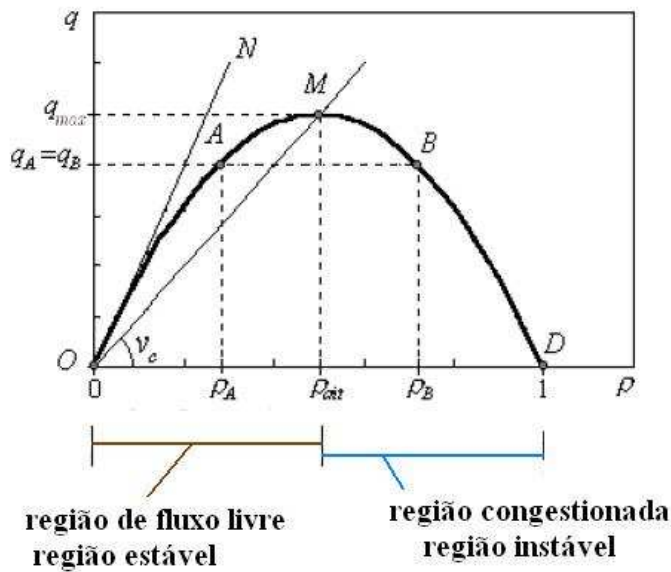


Figura 2.2: Diagrama fundamental da densidade

A modelagem é baseada nas equações de conservação e os modelos matemáticos são representados por equações diferenciais parciais não-lineares. Utilizando a equação da continuidade como a equação da conservação dos veículos temos:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v)}{\partial x} = 0. \quad (2.7)$$

onde ρ denota a densidade de veículos e v a velocidade média dos veículos.

Considerando-se como Q a aceleração referida aos veículos dentro de um volume elementar, tem-se:

$$\frac{\partial v}{\partial t} + \frac{dx}{dt} \frac{\partial v}{\partial x} = \frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} = Q[\rho, v]. \quad (2.8)$$

Para resolver o sistema dado pelas equações (2.7) e (2.8) precisa-se de mais informação sobre Q , para isto podemos ter fechamento por propriedades do fluxo local (equação analítica) ou fechamento por uma equação de evolução para a velocidade (equação funcional).

2.2.1 Modelo de Lighthill-Whitham

O mais antigo e ainda mais popular dos modelos de tráfego macroscópico vem de Lighthill-Whitham (1955) nos artigos sobre ondas cinemáticas. Este modelo aparece também em Richards (1956) que desenvolveu o mesmo modelo, independente do trabalho de Lighthill-Whitham. Como os resultados foram obtidos quase que simultaneamente, o modelo passou a ser conhecido como modelo LWR.

No primeiro artigo (1955), as ondas foram consideradas somente para um sistema de fluxo unidimensional e existe uma relação funcional entre o fluxo q (número de veículos num determinado tempo), a densidade ρ (quantidade por unidade de distância) e a posição x . Utilizando estas relações em conjunto com a equação da continuidade, obteve-se a propriedade desta onda denominada de onda cinemática. Em contraste a teoria clássica, os movimentos da onda clássica serão descritos por ondas dinâmicas que dependem da segunda lei de Newton do movimento ¹

Uma diferença importante é que as ondas cinemáticas possuem apenas uma velocidade de onda em cada ponto, enquanto que as ondas dinâmicas possuem pelo menos duas (para a frente e para trás em relação à média). Considerando a segunda lei de conservação, ou seja, a equação da continuidade.

$$\frac{\partial \rho(t, x)}{\partial t} + \frac{\partial q(t, x)}{\partial x} = 0, \quad (2.9)$$

Tem-se que o fluxo de tráfego q deve ser especificado e isto representa uma dificuldade Lighthill, Whitham e Richards assumiram $q = q_e(\rho(t, x))$ uma relação de equilíbrio entre a

¹Anexo A: anexo 10

densidade e a velocidade v_e , funções convenientemente ajustadas através de dados empíricos. Assim o fluxo é uma função da densidade

$$q(t, x) = q_e(\rho(t, x)) = \rho v_e(\rho(t, x)), \quad (2.10)$$

Então, multiplicando-se (2.9) por

$$c = \left(\frac{\partial q}{\partial \rho} \right)_{x \text{ constante}} = c(\rho, x), \quad (2.11)$$

obtem-se

$$\frac{\partial q}{\partial \rho} \frac{\partial \rho}{\partial t} + \frac{\partial q}{\partial \rho} \frac{\partial q}{\partial x} = 0,$$

$$\frac{\partial q}{\partial t} + c \frac{\partial q}{\partial x} = 0. \quad (2.12)$$

A equação (2.12) representa que q é constante nas ondas que viajam para além do ponto com velocidade c dada por (2.11). Matematicamente, a equação (2.12) implica em um sistema de características (dada por $dx = c dt$, pois $\frac{dx}{dt}$ é a velocidade, neste caso, c), e ao longo de cada uma dessas características o fluxo q é constante.

A velocidade da onda c é a inclinação da curva de concentração do fluxo para um x fixo. Em termos de velocidade média em cada ponto, o fluxo é dado pela equação (2.6) e a velocidade da onda é

$$c = \frac{d}{d\rho}(\rho v) = v + \rho \frac{dv}{d\rho}, \quad (2.13)$$

Assim $c > v$ ² quando a velocidade média aumenta com a concentração (como nos rios, por exemplo, enchentes), enquanto que, $c < v$ ³ no caso em que a velocidade diminuiu com a

²ocorre que $c - v > 0 \rightarrow \rho \frac{dv}{d\rho} > 0$ como sempre tem-se $\rho > 0 \rightarrow \frac{dv}{d\rho} > 0$, ou seja, a velocidade aumenta com a concentração

³ocorre que $c - v < 0 \rightarrow \rho \frac{dv}{d\rho} < 0$ como sempre tem-se $\rho > 0 \rightarrow \frac{dv}{d\rho} < 0$, ou seja, a velocidade decresce com a concentração

concentração (como no fluxo de tráfego).

Ainda, Lighthill, Whitham e Richards substituíram a equação (2.10) na equação (2.9) encontrando

$$\frac{\partial \rho}{\partial t} + \frac{\partial q_e(\rho)}{\partial x} = 0$$

$$\frac{\partial \rho}{\partial t} + \left(\frac{\partial \rho v_e(\rho)}{\partial x} \right) = 0$$

$$\frac{\partial \rho}{\partial t} + \left(v_e \frac{\partial \rho}{\partial x} + \rho \left(\frac{\partial v_e}{\partial \rho} \cdot \frac{\partial \rho}{\partial x} \right) \right) = 0$$

$$\frac{\partial \rho}{\partial t} + \underbrace{\left(v_e + \rho \cdot \frac{\partial v_e}{\partial \rho} \right)}_{C(\rho)} \frac{\partial \rho}{\partial x} = 0.$$

Esta é a equação não linear (Whitham, 1974) que descreve a propagação de ondas cinemáticas com velocidade igual a

$$C(\rho) = v_e(\rho) + \rho \frac{\partial v_e}{\partial \rho} = \frac{dq_e}{d\rho}. \quad (2.14)$$

No artigo de Richards (1956), a teoria é aplicada para o problema de estimar como uma região de concentração crescente (em inglês, *hump*) passa por uma estrada principal lotada. Richards sugere que esta concentração mover-se-á ligeiramente mais lenta do que a velocidade média do veículo, e que os veículos passando por ela terão que reduzir sua velocidade repentinamente (como uma onda de choque) quando entrarem na mesma podendo aumentar a velocidade novamente, de modo gradual, à medida que forem passando. Esta região de grande concentração espalha-se gradualmente ao longo da estrada, e a escala de tempo deste processo foi estimado. O comportamento desta região de concentração ao entrar em um afunilamento (em inglês, *bottleneck*) também é abordado no artigo de Richards.

2.2.2 A equação de Burgers

O modelo de Lighthill-Whitham é muito instrutivo e é a base de uma elaborada teoria de onda de choque (Daganzo, 1999), mas o desenvolvimento de ondas de choque resulta numa séria dificuldade para resolver o modelo de Lighthill-Whitham numericamente. Um método de integração conveniente é o esquema de Godunov (LeVeque, 1990).

Para evitar o desenvolvimento de ondas de choque, Daganzo (1995) adiciona-se um pequeno termo de difusão no modelo de Lighthill-Whitham que suaviza as ondas de choque. O próprio Whitham (1974) sugeriu generalizar a equação (2.10) utilizando uma constante de difusão $D > 0$:

$$q = \rho v_e - D \frac{\partial \rho}{\partial x},$$

cujas derivadas parciais em relação a x é dada por:

$$\frac{\partial q}{\partial x} = \frac{\partial \rho}{\partial x} \left(v_e + \rho \frac{\partial v_e}{\partial \rho} \right) - \frac{\partial}{\partial x} \left(D \frac{\partial \rho}{\partial x} \right). \quad (2.15)$$

Substituindo a equação (2.15) na equação (2.9) tem-se:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho}{\partial x} \left(v_e + \rho \frac{\partial v_e}{\partial \rho} \right) - \frac{\partial}{\partial x} \left(D \frac{\partial \rho}{\partial x} \right) = 0,$$

que resulta na equação

$$\frac{\partial \rho}{\partial t} + \left[v_e(\rho) + \rho \frac{dv_e}{d\rho} \right] \frac{\partial \rho}{\partial x} = \frac{\partial}{\partial x} \left(D \frac{\partial \rho}{\partial x} \right). \quad (2.16)$$

O último termo é denominado de termo de difusão. A constante de difusão D produz uma contribuição significativa somente quando a curvatura $\partial^2 \rho / \partial x^2$ da densidade é grande. No caso de uma densidade máxima (ou seja, um ponto de máximo local ⁴, a curvatura é negativa, que resulta numa redução da densidade. A densidade mínima é suavizada por razões análogas.

⁴ocorre quando $\frac{\partial^2 \rho}{\partial x^2} \frac{\partial^2 \rho}{\partial y^2} - \frac{\partial^2 \rho}{\partial x \partial y} > 0$ e $\frac{\partial^2 \rho}{\partial x^2} < 0$

Para considerações analíticas, Lighthill, Whitham e Richards assumiram a relação linear da densidade-velocidade $v_e(\rho) = v_0 \left(1 - \frac{\rho}{\rho_{eng}}\right)$ sendo ρ_{eng} a densidade ρ do engarrafamento, encontrando

$$\frac{\partial c(t, x)}{\partial t} + c(t, x) \frac{\partial c(t, x)}{\partial x} = D \frac{\partial^2 c(t, x)}{\partial x^2}. \quad (2.17)$$

Esta equação é conhecida como equação de Burgers que contém termos de propagação não-linear e de difusão e pode apresentar uma solução exata devido ao fato de poder ser relacionada com a equação de calor linear,

$$\frac{\partial \psi(t, x)}{\partial t} = D \frac{\partial^2 \psi(t, x)}{\partial x^2}. \quad (2.18)$$

onde ϕ representa o campo de temperaturas. A equação de Burgers supera o problema de ondas de choque, entretanto isto não explica a auto-organização dos engarrafamentos ou ondas de para-e-anda (Helbing, 2001).

2.2.3 Modelos de Prigogine e Phillips

Um modelo alternativo foi proposto por Phillips (1981a, 1981b), o qual deriva da versão modificada do modelo de tráfego de Prigogine da equação do tipo-Boltzmann, que são equações não-lineares íntegro-diferenciais com uma estrutura semelhante à equação de Boltzmann.

Semelhante a Prigogine e Herman (1971), Phillips obteve a equação da continuidade (2.9) e a equação da velocidade V ,

$$\frac{\partial V}{\partial t} + V \frac{\partial V}{\partial x} = -\frac{1}{\rho} \frac{\partial \mathcal{P}}{\partial x} + \frac{1}{\tau(\rho)} (v_e(\rho) - V), \quad (2.19)$$

mas com outro tempo de relaxação $\tau(\rho)$ dependente da densidade. A pressão do tráfego é dada por:

$$\mathcal{P}(\rho, \theta) = \rho \theta, \quad (2.20)$$

sendo $\theta(x, t)$ a variância da velocidade de veículos sendo conduzidos de modo diferente. Para esta variância, um veículo qualquer deriva uma equação dinâmica apropriada ou assume uma relação de densidade-variância tais como $\theta(t, x) = \theta_0 \left(1 - \frac{\rho(t, x)}{\rho_{eng}}\right) \geq 0$ (Phillips, 1981a). Desta forma, a variância decresce com o crescimento da densidade e se anula para a velocidade de equilíbrio $v_e(\rho)$ e para a densidade de engarrafamento ρ_{eng} , como esperado. Entretanto o modelo de Phillips é instável num certo domínio de densidade. Em particular a pressão de tráfego \mathcal{P} cresce com ρ então para altas densidades os veículos poderiam acelerar dentro de regiões congestionadas, o que é irreal.

2.2.4 Modelos de Whitham et al.

Com a equação (??), o modelo de Lighthill-Whitham assume que o fluxo de tráfego está sempre em equilíbrio, ou seja, $q = q_e(\rho)$. Isto é questionável por Kerner (Kerner et al., 2004) para a densidade média, onde não existe relação empírica única entre fluxo e densidade.

Portanto alguns pesquisadores acreditam que a teoria de Lighthill-Whitham é correta, implicando que não existiria tráfego instável mas nos quais os distúrbios são ampliados. Daganzo (1999a) notou que oscilações grandes no fluxo, velocidade e contagem cumulativa crescem em amplitude através de detectores atravessados ao longo de filas de auto-estrada e intervenções sobre rampas.

Whitham (1974) sugeriu uma generalização da teoria de Lighthill-Whitham aplicada a equação (2.19) usando a equação da continuidade com a hipótese de que a adaptação da velocidade é atrasada por um tempo de relaxação τ .

$$\frac{\partial V}{\partial t} + V \frac{\partial V}{\partial x} = \frac{\left(v_e(\rho) - (D/\rho) \frac{\partial \rho}{\partial x} - V \right)}{\tau}.$$

Portanto a equação da velocidade de Whitham compreende

$$\begin{aligned}
\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} \frac{dx}{dt} &= \\
&= \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} V \\
&= \frac{1}{\tau} v_e(\rho) - \frac{1}{\tau} \frac{D}{\rho} \frac{\partial \rho}{\partial x} - \frac{V}{\tau} \\
&= \frac{-1}{\tau} - \left(\frac{D}{\rho} \right) \frac{\partial \rho}{\partial x} + \frac{(v_e(\rho) - V)}{\tau}.
\end{aligned}$$

com constante conveniente $\tau > 0$. De maneira geral os modelos macroscópicos podem ser vistos como casos especiais das equações da continuidade

$$\frac{\partial \rho}{\partial t} + V \frac{\partial \rho}{\partial x} = -\rho \frac{\partial V}{\partial x} + D(\rho) \frac{\partial^2 \rho}{\partial x^2} + \varepsilon_1(t, x) \quad (2.21)$$

e a equação da velocidade

$$\frac{\partial V}{\partial t} + V \frac{\partial V}{\partial x} = -\frac{1}{\rho} \frac{d\mathcal{P}}{d\rho} \frac{\partial \rho}{\partial x} + \nu \frac{\partial^2 V}{\partial x^2} + \frac{1}{\tau} (v_e - V) + \varepsilon_2(t, x) \quad (2.22)$$

As diferenças nestes casos especiais estão nas especificações da difusão $D(\rho)$, das flutuações $\varepsilon_1(t, x)$ e $\varepsilon_2(t, x)$, da pressão de tráfego $\mathcal{P}(\rho)$, da viscosidade $\nu(\rho)$, do tempo de relaxação $\tau(\rho)$ e da velocidade de equilíbrio $v_e(\rho)$ presentes na equação (2.22).

O modelo de Lighthill-Whitham, por exemplo, resulta da equação (2.22) quando multiplica-se a mesma por τ e faz-se o limite $\tau \rightarrow 0$ enquanto que o modelo de Prigogine-Phillips é obtido quando a flutuação $\varepsilon_2(t, x) \rightarrow 0$. Alguns outros modelos consideram parâmetros nulos.

2.3 Descrição cinética

A escala cinética ou mesoscópica assume que o estado do sistema é identificado pela posição e pela velocidade dos veículos. Entretanto sua identificação não refere-se a cada veículo, mas a uma distribuição apropriada de probabilidade sobre o estado microscópico.

Os modelos matemáticos descrevem a evolução de uma função de distribuição por equações íntegro-diferenciais não-lineares com uma estrutura semelhante à equação de Boltzmann que descreve a dinâmica de um gás ideal⁵. Então tem-se

⁵ Anexo A: anexo 11

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} + \frac{F}{m} \frac{\partial f}{\partial u} = \frac{\partial f}{\partial t} \Big|_{colisao} \quad (2.23)$$

sendo f a função de distribuição de probabilidade de uma partícula, F a força externa ao sistema, m a massa da partícula, t o tempo; u a velocidade das partículas e x a posição de cada partícula.

2.3.1 Modelo de Prigogine tipo Boltzmann

Prigogine e Herman (1971) investigaram o modelo cinético para gases (as equações integro-diferenciais tipo Boltzmann) baseados na equação para a densidade no espaço de fase:

$$\tilde{\rho}(t, v, x) = \rho(t, x) \tilde{\mathcal{P}}(t; v, x), \quad (2.24)$$

dada pelo produto da densidade de veículos $\rho(t, x)$ pela distribuição $\tilde{\mathcal{P}}(t, v, x)$ de veículos com velocidade v localizados na posição x e no tempo t . Considerando a conservação do número de veículos, Prigogine e Herman encontraram um tipo de equação da continuidade no espaço de fase:

$$\frac{d_v \tilde{\rho}}{dt} = \frac{\partial \tilde{\rho}}{\partial t} + v \frac{\partial \tilde{\rho}}{\partial x} = \underbrace{\left(\frac{d\tilde{\rho}}{dt} \right)_{acc}}_{\text{comportamento da aceleração}} + \underbrace{\left(\frac{d\tilde{\rho}}{dt} \right)_{int}}_{\text{interação entre veículos}}. \quad (2.25)$$

Sendo que em um sistema de coordenadas movendo-se com velocidade v , a mudança temporal da densidade no espaço de fase é dada pelo comportamento da aceleração e interação dos veículos.

Prigogine sugeriu que o comportamento da aceleração fosse descrito por uma relaxação de uma distribuição da velocidade para alguma distribuição desejada $\tilde{\mathcal{P}}_0(v)$:

$$\left(\frac{d\tilde{\rho}}{dt} \right)_{acc} = \frac{\rho(t, x)}{\tau(\rho(t, x))} \left[\tilde{\mathcal{P}}_0(v) - \tilde{\mathcal{P}}(v, t, x) \right]. \quad (2.26)$$

A quantidade $\tau(\rho)$ novamente denota o tempo de relaxação dependente da densidade. Neste modelo, $\tilde{\mathcal{P}}_0(v)$ reflete a variação da velocidade desejada entre os motoristas, causando a dispersão da velocidade atual v dos veículos. A distribuição $\tilde{\mathcal{P}}_0(v)$ pode ser obtida medindo a distribuição de veículos com grande folga entre eles. A interação entre os veículos é modelada pela equação tipo Boltzmann

$$\begin{aligned} \left(\frac{d\tilde{\rho}}{dt} \right)_{int} &= \int_v^w w > v [1 - \hat{p}(\rho)] |w - v| \tilde{\rho}(t, w, x) \tilde{\rho}(t, v, x) dw \\ &+ \int_0^v w < v [1 - \hat{p}(\rho)] |v - w| \tilde{\rho}(t, v, x) \tilde{\rho}(t, w, x) dw. \end{aligned} \quad (2.27)$$

A equação (2.27) assume que o veículo com velocidade mais rápida w interage com um veículo mais lento com velocidade $v < w$ na razão $|w - v| \tilde{\rho}(t, w, x) \tilde{\rho}(t, v, x)$ que é proporcional à velocidade relativa $|w - v|$ e ao produto da densidade no espaço de fase de interação dos veículos, descrevendo com que frequência veículos com velocidade w e v encontram-se no lugar x . O veículo mais rápido poderá ultrapassar o veículo mais lento com alguma probabilidade $\hat{p}(\rho)$ dependente da densidade. Contudo, a densidade no espaço de fase decresce quando os veículos de velocidade v encontram veículos mais lentos com velocidade $w < v$. Isto é refletido pelo último termo da equação (2.27).

O termo de interação da equação (2.27) pode ser simplificado para

$$\left(\frac{d\tilde{\rho}}{dt} \right)_{int} = [1 - \hat{p}(\rho(t, x))] \rho(t, x) [V(t, x) - v] \tilde{\rho}(t, v, x). \quad (2.28)$$

Pode-se introduzir a velocidade média

$$V(t, x) = \int v \tilde{\mathcal{P}}(t, v, x) dv = \int v \frac{\tilde{\rho}(t, v, x)}{\rho(t, x)} dv. \quad (2.29)$$

e adicionalmente, a variância ⁶ da velocidade

$$\theta(t, x) = \int [v - V(t, x)]^2 \tilde{\mathcal{P}}(t; v, x) dv \quad (2.30)$$

⁶Anexo A: anexo 12

e a velocidade média desejada

$$V_0 = \int \tilde{\mathcal{P}}_0(v) dv.$$

Para a derivada da densidade macroscópica e equação da velocidade, Prigogine e Herman utilizaram o método dos momentos (Kremer, 2003), ou seja, multiplicaram as equações (2.25), (2.26) e (2.28) por 1 e por v depois integraram sobre v obtendo-se as médias. Considerando a densidade de veículos dada por

$$\rho(t, x) = \int \tilde{\rho}(t, v, x) dv. \quad (2.31)$$

Este é um dos métodos que auxiliam na obtenção das equações constitutivas. Assim obtêm-se as equações (2.21) e (2.22) considerando $D(\rho) = 0 = \nu(\rho)$, $\varepsilon_1(t) = 0 = \varepsilon_2(t)$.

Segundo Prigogine e co-pesquisadores existem muitas maneiras de especificar a pressão de tráfego \mathcal{P} que foi adotada pela equação (2.20) e a velocidade de equilíbrio v_e dada por:

$$V_e(\rho, \theta) = V_0 - \tau(\rho)[1 - \hat{p}(\rho)]\rho\theta \quad (2.32)$$

Outros pesquisadores presumem que estas representações são obtidas por considerações fenomenológicas. Infelizmente, as relações (2.20) e (2.32) são válidas somente para baixas densidades, mas teorias para altas densidades também estão disponíveis.

2.3.2 Modelo de Paveri Fontana (MPF)

Paveri Fontana (1975) prosseguiu uma investigação muito detalhada do modelo de tráfego de Prigogine da cinética dos gases e reconheceram algumas características. Ele criticou o fato de que o termo de aceleração (2.26) descreveria uma velocidade descontínua com saltos de descontinuidade ocorrendo com a razão proporcional para $\frac{1}{\tau(\rho)}$. Além disso, a velocidade desejada dos veículos era uma propriedade da estrada e não dos motoristas, enquanto existem atualmente diferentes personalidades de motoristas: agitados, que dirigem rápido, tímidos, que dirigem devagar (Daganzo, 1995).

Paveri Fontana resolveu estes problemas distinguindo os diferentes tipos de veículo-motorista. Estes foram caracterizados pela velocidade individual desejada v_0 . Como consequência introduziu-se uma extensão da densidade no espaço de fase $\rho_*(t, v, v_0, x)$ e uma equação correspondente a da cinética dos gases. Por integração sobre v_0 , Paveri Fontana obteve a equação (2.25). Enquanto Paveri Fontana especificou o termo de interação análogo ao de Prigogine, uma diferente aproximação foi escolhida para o termo de aceleração. Isto corresponde a lei de aceleração microscópica

$$\frac{dv}{dt} = \frac{v_0 - v}{\tau(\rho)}, \quad (2.33)$$

analogamente resultando na fórmula

$$\left(\frac{d\tilde{\rho}}{dt}\right)_{acc} = \frac{-\partial}{\partial v} \left[\tilde{\rho}(t, v, x) \frac{\tilde{V}_0(t, v, x) - v}{\tau(\rho(t, x))} \right] \quad (2.34)$$

(Helbing, 1995).

Aqui a quantidade $\tilde{V}_0(t, v, x)$ denota a velocidade média desejada dos veículos movendo-se com velocidade v e é dada por $\tilde{V}_0(t, v, x) = \int dv_0 v_0 \rho_*(t, v, v_0, x) / \tilde{\rho}(t, v, x) \sim V_0(t, x) + [v - V(t, x)]C'(t, x)/\theta(t, x)$, que é maior para veículos mais rápidos. A velocidade média desejada é representada por $V_0(t, x) = \int dv_0 \int dv v_0 \rho_*(t, v, v_0, x) / \rho(t, x)$ no espaço x e no tempo t , e $C'(t, x) = \int dv_0 \int dv (v - V)(v_0 - V_0) \rho_*(t, v, v_0, x) / \rho(t, x)$ representa a covariância entre a velocidade atual e a velocidade desejada. A respeito destas diferenças, a equação macroscópica para a densidade e a velocidade média concordam exatamente com aquelas de Prigogine, contudo, as equações para a variância da velocidade θ , a covariância (σ^2)⁷, a velocidade média desejada V_m e consequentes são diferentes.

A construção e propriedades da solução para a equação de Paveri Fontana tem sido cuidadosamente estudada por Barone (1981) e Semenzato (1981a, 1981b). Observa-se que existem várias alternativas para o termo de aceleração da equação (2.34).

⁷Anexo A: anexo 13

2.3.2.1 Modelo para Velocidades Previstas Especificadas para MPF

Com o objetivo de se corrigir algumas deficiências do modelo de tráfego obtido através da teoria cinética dos gases proposta por Prigogine (Prigogine, 1971), ou seja, equação cinética de tráfego, Paveri Fontana desenvolveu um tratamento do tipo-Boltzmann para o fluxo de tráfego levando-se em conta o comportamento individual do carro, ou seja, sua aceleração. No modelo de Paveri Fontana o estado do tráfego é caracterizado pela função de distribuição de um veículo $g(x, c, w, t)$ tal que dado um tempo t , $g(x, c, w, t)dx dc dw$ fornece o número de veículos numa estrada entre as posições x e $x + dx$ e velocidade atual entre c e $c + dc$, com velocidade desejada entre w e $w + dw$. Para uma estrada de mão única, após ocorrer uma ultrapassagem, a função de distribuição $g(x, c, w, t)$ satisfaz a seguinte equação:

$$\begin{aligned} & \frac{\partial g}{\partial t} + c \frac{\partial g}{\partial x} + \frac{\partial}{\partial c} \left(g \frac{\partial c}{\partial t} \right) + \frac{\partial}{\partial w} \left(g \frac{\partial w}{\partial t} \right) \\ &= \underbrace{f(x, c, t) \int_c^\infty (1 - p)(c' - c)g(x, c', w, t)dc'}_{(I)} \\ & \quad - \underbrace{g(x, c, w, t) \times \int_0^c (1 - p)(c - c')f(x, c', t)dc'}_{(II)}, \end{aligned} \quad (2.35)$$

denominada equação cinética de Paveri Fontana, na qual

$$f(x, c, t) = \int_0^\infty g(x, c, w, t)dw \quad (2.36)$$

é a função de distribuição da velocidade de um veículo.

Na equação (2.35) temos o item (I) correspondente à situação do veículo com velocidade c' que deve desacelerar para a velocidade c causando um aumento na função de distribuição. O item (II) descreve um decrescimento na função de distribuição devido à situação em que um veículo com velocidade c deve desacelerar para uma velocidade menor do que a velocidade c' , basicamente considera que os veículos mais lentos podem ser imediatamente ultrapassados com uma probabilidade p .

Assumindo-se que os motoristas aproximam-se exponencialmente da velocidade desejada num tempo com uma constante de relaxação de tempo τ , pode-se escrever:

$$\frac{dc}{dt} = \frac{w - c}{\tau} \quad \text{e} \quad \frac{dw}{dt} = 0. \quad (2.37)$$

A lei da aceleração dada pela equação (2.35) representa uma boa aproximação, desde que os motoristas reduzam gradualmente a aceleração após alcançar a velocidade desejada.

A equação (2.35) é uma equação íntegro-diferencial denominada equação cinética de Pavari Fontana e apresentando uma grande dificuldade de se encontrar uma solução analítica para os quais os processos de interação não podem ser abandonados. Para superar esta dificuldade integra-se a equação 2.35 com respeito a velocidade desejada, obtendo-se uma equação conhecida por equação reduzida de Pavari Fontana:

$$\frac{\partial f}{\partial t} + c \frac{\partial f}{\partial x} + \frac{\partial}{\partial c} \left(f \frac{V_0 - c}{\tau} \right) = f(x, c, t) \int_0^\infty (1 - p)(c' - c) f(x, c', t) dc' \quad (2.38)$$

na qual

$$V_0(x, c, t) = \int_0^\infty w \frac{g(x, c, w, t)}{f(x, c, t)} dw, \quad (2.39)$$

representa a velocidade média desejada.

Ao se considerar o tráfego homogêneo e estacionário, a velocidade média desejada para os veículos é diretamente proporcional a velocidade atual c . Sendo esta relação dada por:

$$V_0(x, c, t) = V_0(c) = kc, \quad (2.40)$$

ou seja, esta velocidade será independente do tempo e da posição na qual o veículo se encontra na estrada k é uma constante. Este é um caso particular e simplifica a equação (2.38) para a velocidade média desejada.

Sabe-se que a função de distribuição deve satisfazer as seguintes condições de contorno

$$\lim_{w \rightarrow 0} g(x, c, w, t) = 0 \quad \text{e} \quad \lim_{w \rightarrow \infty} g(x, c, w, t) = 0, \quad (2.41)$$

o que torna a equação (2.38) igual a

$$\frac{\partial f}{\partial t} + c \frac{\partial f}{\partial x} + \frac{\partial}{\partial c} \left(f \frac{kc - c}{\tau} \right) = \mathcal{C}(c, c'), \quad (2.42)$$

sendo que $\mathcal{C}(c, c')$ representa o termo de colisão, ou seja, ganhos menos perdas.

Denominando-se por $\beta = \frac{k-1}{\tau}$, na qual τ é constante, então a equação (2.42) transforma-se em

$$\frac{\partial f}{\partial t} + c \frac{\partial f}{\partial x} + \beta \frac{\partial(cf)}{\partial c} = \mathcal{C}(c, c'). \quad (2.43)$$

O modelo de Paveri Fontana (MPF) resolveu parcialmente o problema dos diferentes tipos de veículo-motorista, os quais foram caracterizados pela velocidade individual desejada v_0 . Como consequência introduziram uma extensão da densidade no espaço de fase $\rho_*(t, v, v_0, x)$ e uma equação correspondente a da cinética dos gases.

As equações reduzidas de Paveri Fontana para modelos com valores especificados da velocidade prevista para uma pista foi o primeiro estudo deste trabalho. Utilizou-se o artigo de Velasco et al. (2005) para os dados iniciais assim como o método das características e o método das diferenças finitas explícito, considerando-se um problema de condições de contorno periódicas (Prado et al., 2006).

O método das características mostrou-se adequado para solucionar o sistema de equações quase-lineares que governam o fluxo de tráfego simulado e em geral não apresenta grandes dificuldades para ser implementado computacionalmente quando usado para solucionar problemas de fluxo unidimensional, como o caso do problema investigado. Outro problema estudado foi para uma pista com entrada lateral (Prado et al., 2007) utilizando o método das diferenças finitas explícito, o que até o presente momento não havia sido feito.

3 *Método das características e método das diferenças finitas*

Neste capítulo apresenta-se os métodos das características e das diferenças finitas utilizados na resolução das equações reduzidas de Paveri Fontana na escala macroscópica. Abordou-se dois problemas: uma pista circular e uma pista com entrada lateral. Para uma pista utilizou-se o método das características (Prado et al., 2006) e o método das diferenças finitas explícito. Para uma pista com entrada lateral (Prado et al., 2007) utilizou-se o método das diferenças finitas explícito. As variáveis são a densidade veicular e a velocidade média do tráfego

3.1 Equações macroscópicas do tráfego

O método utilizado na dedução das equações macroscópicas é semelhante ao desenvolvido por Velasco (Velasco et al., 2005). Por se tratar de um problema não fechado, existem vários modos de se obter os resultados. Usa-se os valores encontrados para a Pressão do Tráfego \mathcal{P} e uma condição para a velocidade média desejada \mathcal{W} . Os cálculos numéricos são realizados com base no método das características (Streeter, 1997).

Nas equações do tráfego reduzidas de Paveri Fontana, as quantidades macroscópicas ρ e v são funções da função de distribuição da velocidade de um veículo. Para qualquer função de distribuição $f(x, c, t)$, a densidade veicular ρ e a velocidade média v são definidas através de:

$$\rho(x, t) = \int_0^{\infty} f(x, c, t) dc \quad (3.1)$$

e

$$v(x, t) = \int_0^\infty c \frac{f(x, c, t)}{\rho(x, t)} dc. \quad (3.2)$$

respectivamente.

A integração da equação (2.38), conhecida por equação reduzida do tráfego de Paveri Fontana, sobre todos os atuais valores da velocidade c , produz a equação da continuidade

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho v}{\partial x} = 0 \quad (3.3)$$

e a equação da velocidade como sendo

$$\rho \left(\frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} \right) + \frac{\partial \mathcal{P}}{\partial x} = \rho \frac{\mathcal{W} - v}{\tau} - \rho(1 - p)\mathcal{P} \quad (3.4)$$

obtida pela multiplicação da equação (2.38) por c e efetuando a integral sobre todos os valores da velocidade atual.

Na equação (3.4) foram introduzidas a velocidade média prevista definida por

$$\mathcal{W} = \int_0^\infty v(x, c, t) \frac{f(x, c, t)}{\rho(x, t)} dc \quad (3.5)$$

e a pressão do tráfego

$$\mathcal{P} = \int_0^\infty (c - v)^2 f(x, c, t) dc, \quad (3.6)$$

que relaciona-se com a variância $\Theta(x, t)$ da velocidade definida por $\Theta = \frac{1}{\rho} \int_0^\infty (c - V)^2 f(x, c, t) dc$ ¹, através da equação (2.20)

$$\mathcal{P} = \rho(x, t)\Theta(x, t).$$

As equações são válidas desde que a função de distribuição $f(x, c, t)$ satisfaça as condições

¹É importante observar que a variância mede a dispersão (ou espalhamento) dos dados em torno da média v .

$$\lim_{c \rightarrow 0} f(x, c, t) = 0 \quad \text{e} \quad \lim_{c \rightarrow \infty} f(x, c, t) = 0. \quad (3.7)$$

Usa-se como hipóteses de que \mathcal{W} é constante ($= V_0$) e que a pressão do tráfego é diretamente proporcional à densidade veicular e ao quadrado da velocidade média ².

$$p = 1 - \frac{\rho}{\hat{\rho}}, \quad \text{e} \quad \mathcal{P} = \frac{\rho v^2}{\alpha} \quad (3.8)$$

sendo α uma constante adimensional. Soluciona-se as equações (2.7) e (3.4) pelo método das características e pelo método das diferenças finitas explícito.

Para o caso da pista com entrada lateral simula-se o fluxo de tráfego pelo método de diferenças finitas explícito utilizando-se as seguintes equações da continuidade e da velocidade, respectivamente:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho v}{\partial x} = q \quad (3.9)$$

e

$$\rho \left(\frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} \right) + \frac{\partial \mathcal{P}}{\partial x} = \rho \frac{\mathcal{W} - v}{\tau} - \rho(1 - p)\mathcal{P} - qv. \quad (3.10)$$

Sendo que q dado em veículos/hora/km representando o fluxo de entrada de veículos por unidade de largura da via transversal.

3.2 Método das características (mc)

Considera-se o problema de valor inicial, para a equação de primeira ordem quase linear, dadas as funções $a(x, t, u)$, $b(x, t, u)$ e $d(x, t, u)$ cujo objetivo é encontrar $u(x, t)$:

²No artigo de Velasco et al. utilizou-se o método de Chapman-Enskog para encontrar as aproximações para f e com isto a aproximação de ordem-zero determina a pressão do tráfego proporcional ao quadrado da velocidade média.

$$\begin{cases} a(x, t)u_x + b(x, t)u_t + d(x, t) = 0, & t > 0, -\infty < x < \infty, \\ u(x, 0) = u_0(x). \end{cases} \quad (3.11)$$

A equação diferencial de primeira ordem (3.11), definida inicialmente no sistema de coordenadas (x, t) será reescrita em um outro sistema de coordenadas (s, t) definido convenientemente. Sejam as curvas $x(s)$ e $t(s)$ definidas por:

$$\begin{aligned} \frac{dx}{ds} &= a(x(s), t(s), u), & x(0) &= x_i, \\ \frac{dt}{ds} &= b(x(s), t(s), u), & t(0) &= t_i, \end{aligned} \quad (3.12)$$

estas são denominadas curvas características no espaço (x, t, u) . Aplicando a regra da cadeia para se derivar $u(x(s), t(s))$, ao longo das características, tem-se

$$\frac{du}{ds} = u_x \frac{dx}{ds} + u_t \frac{dt}{ds} = a(x, t, u)u_x + b(x, t, u)u_t. \quad (3.13)$$

Comparando a equação (3.13) com a equação (3.11) obtem-se

$$\begin{cases} \frac{du}{ds} = -d(x(s), t(s), u), \\ u(0) = u_0(x_i). \end{cases} \quad (3.14)$$

isto é, define-se uma equação diferencial ordinária no novo sistema de coordenadas (s, t) .

Comparando as equações (3.11) e (3.14), observa-se que a equação diferencial parcial transformou-se em um sistema de equações diferenciais ordinárias.

Assim a solução será encontrada em pontos que estão sobre as características. Um processo de interpolação pode ser utilizado se estas características se afastam em algumas regiões. Este método é considerado um método para solução analítica de equações diferenciais parciais de primeira ordem.

O método das características pode ser aplicado para solucionar sistemas de duas equações diferenciais parciais de primeira ordem quase lineares e hiperbólicas.

Sejam L_1 e L_2 equações contendo as variáveis dependentes, ρ , a densidade veicular, e v , a velocidade média. Essas equações serão combinadas por meio de um multiplicador desconhecido λ , como $L = L_1 + \lambda L_2$. Quaisquer valores reais distintos de λ fornecem duas equações em ρ e V que representam o mesmo fenômeno físico que as duas equações originais, L_1 e L_2 , e que podem substituí-las diante de qualquer solução. Substituindo L_1 e L_2 na expressão de L , obtém-se, após um rearranjo, as equações que representam as derivadas totais, $\frac{d\rho}{dt}$ e de $\frac{dV}{dt}$, respectivamente.

Atribui-se L_2 a equação da continuidade (2.7) e L_1 a equação da velocidade (3.4),

$$L_2 = \frac{\partial \rho}{\partial t} + \rho \frac{\partial v}{\partial x} + v \frac{\partial \rho}{\partial x} = 0. \quad (3.15)$$

e

$$L_1 = \rho \left(\frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} \right) + \frac{\partial \mathcal{P}}{\partial x} - \left(\rho \frac{\mathcal{W} - v}{\tau} + \rho(1 - p)\mathcal{P} \right). \quad (3.16)$$

Substituindo-se na equação da velocidade (3.4), $\mathcal{W} = V_0$ e a pressão do tráfego dada pela equação (??), encontra-se

$$\rho \left(\frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial x} \left(\frac{\rho v^2}{\alpha} \right) = \rho \frac{v_0 - v}{\tau} - \rho(1 - p) \frac{\rho v^2}{\alpha}. \quad (3.17)$$

Na equação (3.17), ρ e v são funções de x e α é uma constante adimensional dada por

$$\alpha = \frac{\rho_e (1 - p) v_e}{\beta}, \quad (3.18)$$

na qual ρ_e e v_e referem-se ao estado estacionário.

Desenvolvendo a parcial com relação a x na equação (3.15) e agrupando as derivadas parciais da mesma função e aplicando o método das características, ou seja, $L = L_1 + \lambda L_2$, encontra-se

$$L_1 = \frac{\partial v}{\partial t} + \frac{\partial v}{\partial x} v_{c(L_1)} \quad (3.19)$$

e

$$L_2 = \frac{\partial \rho}{\partial t} + \frac{\partial \rho}{\partial x} v_{c(L_2)} \quad (3.20)$$

sendo genericamente $v_c = \frac{dx}{dt}$.

Com a condição de que $v_{c(L_1)} = v_{c(L_2)}$, a equação abaixo terá que ser satisfeita

$$\rho \lambda^2 + \frac{2v}{\alpha} \lambda - \frac{v^2}{\rho \alpha} = 0, \quad (3.21)$$

que representa uma equação do segundo grau em λ . Para o caso deste trabalho, tem-se duas raízes reais e distintas para λ ,

$$\lambda = \frac{v}{\alpha \rho} (-1 \pm \sqrt{1 + \alpha}), \quad (3.22)$$

ou seja, solução do tipo hiperbólico.

Substituindo a equação (3.22) em $L = L_1 + \lambda L_2$, encontram-se quatro equações diferenciais ordinárias

$$C^+ \left\{ \begin{array}{l} \frac{dv}{dt} + \frac{v}{\alpha \rho} (-1 + \sqrt{1 + \alpha}) \frac{d\rho}{dt} + (1 - p) \frac{\rho v^2}{\alpha} - \frac{(V_0 - v)}{\tau} = 0, \\ \frac{dx}{dt} = v - \frac{v}{\alpha} (-1 + \sqrt{1 + \alpha}), \end{array} \right. \quad (3.23)$$

e também

$$C^- \left\{ \begin{array}{l} \frac{dv}{dt} + \frac{v}{\alpha \rho} (-1 - \sqrt{1 + \alpha}) \frac{d\rho}{dt} + (1 - p) \frac{\rho v^2}{\alpha} - \frac{(V_0 - v)}{\tau} = 0, \\ \frac{dx}{dt} = v - \frac{v}{\alpha} (-1 - \sqrt{1 + \alpha}). \end{array} \right. \quad (3.24)$$

As equações (3.23) e (3.24) são as equações do método das características, nas quais C^+ e C^- representam as curvas características positivas e negativas, respectivamente. Ambas pos-

suem inclinações positivas pois o fluxo de tráfego é maior do que a velocidade das ondas sonoras (Sivaloganathan, 1974) e (Sivaloganathan, 1978). Conhecidas as condições iniciais e de contorno pode-se traçar a malha das curvas características.

3.2.1 O processo geral do método das características

Devido à inclinação das características começa-se o processo de discretização da malha da condição de contorno para a posição do trecho $0 \leq x \leq L$, onde L é a extensão da estrada.

O esquema deste processo, denominado esquema das distâncias especificadas do método das características (Sivaloganathan, 1974), diferenciado do esquema da malha retangular (Sivaloganathan, 1978), está representado na Figura 3.3 na qual:

- (i) Divide-se o comprimento da estrada em n partes iguais de comprimento Δx .
- (ii) Inicia-se pela condição de contorno em A , $x = L$, com $P_{1C_1} = (L, 0)$, traçando-se uma curva característica positiva C_1 .
- (iii) Em B , $x = L - \Delta x$, com $P_{1C_2} = (L - \Delta x, 0)$, traça-se uma curva característica positiva C_2 e em AB , ponto médio de A e B , traça-se uma curva de característica negativa. A intersecção destas curvas ocorre no ponto M .
- (iv) Em A traça-se uma curva característica negativa, encontrando-se o ponto N sobre a curva C_2 . Interpola-se os pontos M e N encontrando o ponto P_{2C_2} .
- (v) Em C , $x = L - 2\Delta x$, com $P_{1C_3} = (L - 2\Delta x, 0)$, traça-se uma curva característica positiva C_3 e em BC , ponto médio de B e C , traça-se uma curva característica negativa. A intersecção destas curvas ocorre no ponto M^* .
- (vi) Em B traça-se uma curva característica negativa, encontrando-se o ponto P_{2C_3} .
- (vii) Em N traça-se uma curva característica negativa encontrando-se N^* sobre a curva C_3 . Interpola-se os pontos P_{2C_3} e N^* encontrando o ponto P_{3C_3} .

Passo 1

Atribuindo-se $b_{\pm} = \frac{-1 \pm \sqrt{\alpha + 1}}{\alpha}$ e $q_{\pm} = \frac{(\alpha + 1) \pm \sqrt{\alpha + 1}}{\alpha}$ as equações (3.23) e (3.24), com o uso de (3.8) serão escritas na forma:

$$\frac{1}{v} \frac{dv}{dt} + \frac{b_{\pm}}{\rho} \frac{d\rho}{dt} = \underbrace{\frac{V_0 - v}{\tau v} - \frac{\rho v}{\alpha} \left(1 - \frac{\rho}{\hat{\rho}}\right)}_{f(v, \rho)}, \quad (3.25)$$

e

$$\frac{dx}{dt} = v q_{\pm}, \quad (3.26)$$

respectivamente.

Passo 2

Calcula-se $f_L(v_L, \rho_L)$ e $f_R(v_R, \rho_R)$ sendo conhecidos a velocidade e a densidade em L e R através das equações:

$$f_L(v_L, \rho_L) = \frac{V_0 - v_L}{\tau v_L} - \frac{\rho_L v_L}{\alpha} \left(1 - \frac{\rho_L}{\hat{\rho}}\right) \quad (3.27)$$

e

$$f_R(v_R, \rho_R) = \frac{V_0 - v_R}{\tau v_R} - \frac{\rho_R v_R}{\alpha} \left(1 - \frac{\rho_R}{\hat{\rho}}\right) \quad (3.28)$$

Passo 3

Inicia-se, como primeira aproximação, fazendo-se a determinação do ponto M como

$$t_M = \frac{t_L + t_R}{2}, \quad x_M = \frac{x_L + x_R}{2}, \quad \rho_M = \frac{\rho_L + \rho_R}{2} \quad \text{e} \quad v_M = \frac{v_L + v_R}{2}. \quad (3.29)$$

Passo 4

Calculam-se novas aproximações das variáveis no ponto M do item anterior, denotadas por

$$t_M^*, \quad x_M^*, \quad \rho_M^*, \quad v_M^*.$$

- Cálculo de t_M^* e x_M^* .

Integrando a equação (3.26), obtém-se um sistema para x_M e t_M formado pelas equações

$$x_M - x_L = \frac{q_+}{2}(v_M + v_L)(t_M - t_L) \quad (3.30)$$

e

$$x_M - x_R = \frac{q_-}{2}(v_M + v_R)(t_M - t_R); \quad (3.31)$$

Denominando

$$a_L = \frac{q_+}{2}(v_M + v_L) = \frac{(\alpha + 1) + \sqrt{\alpha + 1}}{\alpha}(v_M + v_L) \quad (3.32)$$

e

$$a_R = \frac{q_-}{2}(v_M + v_R) = \frac{(\alpha + 1) - \sqrt{\alpha + 1}}{\alpha}(v_M + v_R), \quad (3.33)$$

reescreve-se as equações (3.30) e (3.31), respectivamente por:

$$x_M^* - x_L = a_L(t_M^* - t_L) \quad (3.34)$$

e

$$x_M^* - x_R = a_R(t_M^* - t_R), \quad (3.35)$$

no qual chamou-se x_M por x_M^* e t_M por t_M^* para denotar que se está calculando uma nova aproximação para estas variáveis. Resolvendo o sistema descrito pelas equações (3.34) e (3.35) encontra-se

$$t_M^* = \frac{x_R - x_L + a_L t_L - a_R t_R}{a_L - a_R} \quad (3.36)$$

e

$$x_M^* = x_L + a_L(t_M^* - t_L). \quad (3.37)$$

- Cálculo de ρ_M^* e v_M^* .

Integrando a equação (3.25) obtém-se um sistema para ρ_M e v_M dados pelas equações:

$$b_+(\ln \rho_M - \ln \rho_L) + \ln v_M - \ln v_L = \left[\frac{f_M(v_M, \rho_M) + f_L(v_L, \rho_L)}{2} \right] (t_M^* - t_L) \quad (3.38)$$

e

$$b_-(\ln \rho_M - \ln \rho_R) + \ln v_M - \ln v_R = \left[\frac{f_M(v_M, \rho_M) + f_R(v_R, \rho_R)}{2} \right] (t_M^* - t_R), \quad (3.39)$$

na qual utilizou-se para f_M a definição dada na equação (3.27), substituindo M por L .

Denominando

$$b_L = \left[\frac{f_M(v_M, \rho_M) + f_L(v_L, \rho_L)}{2} \right] (t_M^* - t_L) \quad (3.40)$$

e

$$b_R = \left[\frac{f_M(v_M, \rho_M) + f_R(v_R, \rho_R)}{2} \right] (t_M^* - t_R), \quad (3.41)$$

as equações (3.38) e (3.39), para $\rho_M^* = \rho_M$ e $v_M^* = v_M$, formam o sistema

$$b_+(\ln \rho_M^* - \ln \rho_L) + \ln v_M^* - \ln v_L = b_L \quad (3.42)$$

e

$$b_-(\ln \rho_M^* - \ln \rho_R) + \ln v_M^* - \ln v_R = b_R, \quad (3.43)$$

cuja solução é dada por

$$\rho_M^* = \exp \left\{ [b_L - b_R + \ln(v_L) - \ln(v_R) + \frac{\sqrt{\alpha+1}-1}{\alpha} \ln(\rho_L) + \frac{\sqrt{\alpha+1}+1}{\alpha} \ln(\rho_R)] \frac{\alpha}{2\sqrt{\alpha+1}} \right\} \quad (3.44)$$

e

$$v_M^* = \exp \left\{ b_L + \ln(v_L) + \frac{\sqrt{\alpha+1}-1}{\alpha} [\ln(\rho_L) - \ln(\rho_M^*)] \right\}. \quad (3.45)$$

Passo 5

Com os valores da aproximação anterior e com os valores da aproximação do passo 4, verifica-

se a convergência através das condições:

$$|x_M - x_M^*| < \varepsilon x_M, \quad |t_M - t_M^*| < \varepsilon t_M, \quad |v_M - v_M^*| < \varepsilon v_M, \quad \text{e} \quad |\rho_M - \rho_M^*| < \varepsilon \rho_M, \quad (3.46)$$

nas quais ε representa a precisão da convergência. No programa numérico adotou-se $\varepsilon = 10^{-7}$. Se as condições dadas em (3.46) forem satisfeitas, a determinação do ponto M foi encontrada. Caso contrário, atribui-se

$$x_M = x_M^*, \quad t_M = t_M^*, \quad v_M = v_M^* \quad \text{e} \quad \rho_M = \rho_M^* \quad (3.47)$$

e volta-se ao passo 4, repetindo este processo, até se obter a precisão desejada.

3.3 Simulação numérica do método das características

Inicialmente tem-se um tráfego homogêneo e estacionário (indicado pelo índice e), com densidade e velocidade iguais a

$$\rho_e = 28 \text{ veículos/km} \quad \text{e} \quad v_e(\rho_e) = 84 \text{ km/h} \quad (3.48)$$

que sofre uma pequena perturbação periódica na velocidade média (Helbing, 1996) representando alguns veículos movendo-se mais rapidamente e outros mais lentamente. Assim, as condições iniciais consideradas são:

$$\rho_0 = 28 \text{ veículos/km} \quad \text{e} \quad v_0 = v(x, 0) = 84 + 0,84 \sin\left(\frac{2\pi x}{L}\right), \quad (3.49)$$

com $0 \leq x \leq L$ dados em km. Implementa-se para o método numérico a extensão da estrada igual a $L = 12$ km e a constante de relaxação de tempo como $\tau = \frac{1}{120}h$. Adota-se $\hat{\rho} = 140$ veículos/km e $\alpha = 100$, obtido a partir de resultados experimentais (Velasco et al., 2005).

O sistema de equações diferenciais ordinárias (3.23) e (3.24) requerem a especificação das

condições iniciais e de contorno. Com o objetivo de simplificar a análise, serão consideradas condições de contorno periódicas para a densidade veicular $\rho(0, t) = \rho(L, t)$ e a velocidade média $V(0, t) = V(L, t)$, no qual L representa a extensão da estrada estudada.

Considerou-se os dados do artigo (Velasco et al., 2005) para fazer uma comparação entre os métodos de resolução para uma validação dos mesmos, o programa (1) utilizado está no apêndice A.

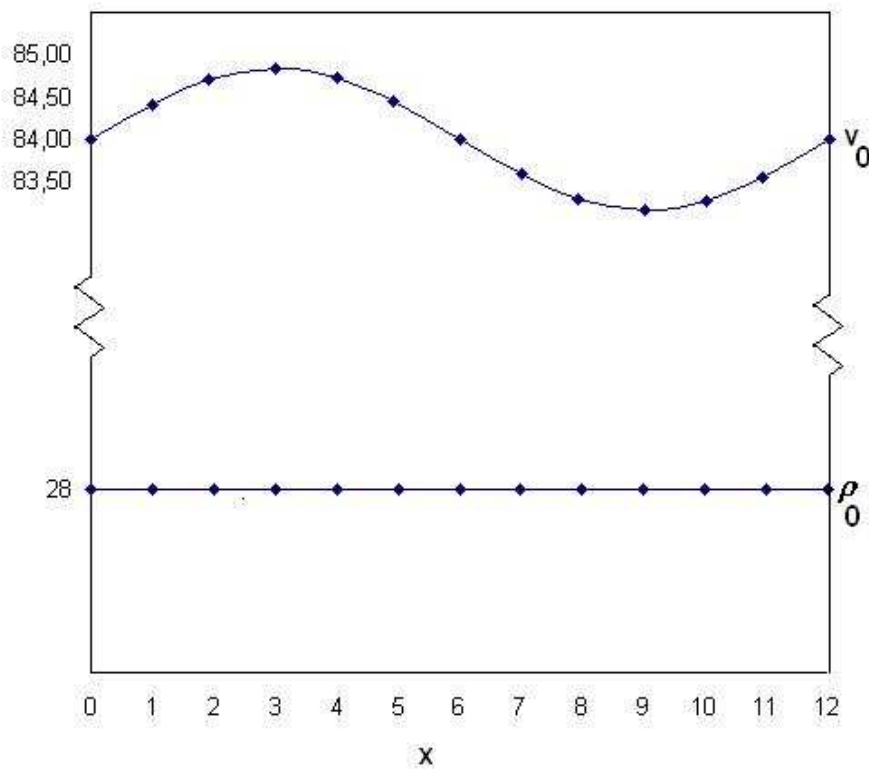


Figura 3.5: Valores da condição inicial

Utilizando-se a equação (3.18), calcula-se β , a qual fornecerá a constante k (da equação 2.40) e consequentemente v_0 . Para tempos posteriores, na equação (3.18) se utiliza a equação (3.49) para a expressão da velocidade e repete-se o processo para se obter v_0 .

Na tabela 3.1 apresenta-se a evolução da velocidade média v em função do trecho da estrada x para alguns valores de tempo t escolhidos para a simulação considerando-se a divisão do trecho da estrada em 200 partes iguais, $n = 200$.

Os resultados apresentados na tabela 3.1 ilustram uma pequena amostra dos 266.112 pontos calculados, considerando-se $n = 200$, para simular uma hora do movimento.

Tabela 3.1: Valores de $v(x, t)$ para a posição x e para nove valores de t .

| x | $t = 0$ | $t = 0,2$ | $t = 0,3$ | $t = 0,4$ | $t = 0,5$ | $t = 0,7$ | $t = 0,8$ | $t = 0,9$ | $t = 1,0$ |
|-----|---------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 0 | 84,0000 | 80,9412 | 80,9572 | 80,9236 | 80,9542 | 80,9277 | 80,9608 | 80,9316 | 80,9394 |
| 1 | 84,4050 | 80,9505 | 80,9495 | 80,9251 | 80,9598 | 80,9350 | 80,9600 | 80,9257 | 80,9483 |
| 2 | 84,7090 | 80,9579 | 80,9401 | 80,9309 | 80,9612 | 80,9441 | 80,9550 | 80,9240 | 80,9558 |
| 3 | 84,8400 | 80,9616 | 80,9302 | 80,9407 | 80,9574 | 80,9536 | 80,9455 | 80,9276 | 80,9605 |
| 4 | 84,7360 | 80,9599 | 80,9247 | 80,9500 | 80,9499 | 80,9594 | 80,9361 | 80,9347 | 80,9600 |
| 5 | 84,4500 | 80,9540 | 80,9236 | 80,9574 | 80,9406 | 80,9611 | 80,9284 | 80,9436 | 80,9552 |
| 6 | 84,0000 | 80,9440 | 80,9280 | 80,9613 | 80,9307 | 80,9576 | 80,9240 | 80,9531 | 80,9460 |
| 7 | 83,5950 | 80,9346 | 80,9357 | 80,9600 | 80,9250 | 80,9504 | 80,9251 | 80,9590 | 80,9366 |
| 8 | 83,2910 | 80,9271 | 80,9451 | 80,9544 | 80,9237 | 80,9411 | 80,9305 | 80,9609 | 80,9288 |
| 9 | 83,1600 | 80,9233 | 80,9547 | 80,9445 | 80,9279 | 80,9312 | 80,9398 | 80,9578 | 80,9243 |
| 10 | 83,2640 | 80,9251 | 80,9602 | 80,9351 | 80,9354 | 80,9253 | 80,9488 | 80,9508 | 80,9252 |
| 11 | 83,5500 | 80,9312 | 80,9614 | 80,9275 | 80,9446 | 80,9239 | 80,9563 | 80,9416 | 80,9303 |
| 12 | 84,0000 | 80,9412 | 80,9572 | 80,9236 | 80,9542 | 80,9277 | 80,9608 | 80,9316 | 80,9394 |

Fonte: Resultados do programa(1) para a velocidade v pelo método das características.

Para a densidade ρ nos mesmos tempos utilizados para a velocidade observa-se na Figura 3.3 a oscilação é uniforme em quase todos os tempos mostrados, apresentando uma oscilação maior no tempo $t = 1,0$.

3.4 Aspectos conclusivos para o método das características

Nesta seção apresentam-se resultados obtidos da simulação do fluxo de tráfego para condições de contorno periódicas tanto para a densidade veicular, ρ , como para a velocidade média v , como exemplo, uma pista circular. Os cálculos numéricos foram realizados usando-se o denominado esquema das distâncias especificadas do método das características. Por meio deste esquema são determinados os resultados das variáveis dependentes, v e ρ , para as curvas características positivas que emanam em pontos fixados no eixo x . Como o fluxo de tráfego é maior do que a velocidade das ondas sonoras (fluxo super-crítico), as curvas características positivas e negativas têm inclinações positivas, a primeira característica examinada emana do ponto $x = L$, a segunda de $x = L - \Delta x$, e assim sucessivamente. Em geral o esquema do método das características das distâncias especificadas, quando comparado com o esquema da malha retangular, utiliza menor tempo de processamento embora ocupe maior espaço de memória do computador. Com os resultados obtidos pelo esquema adotado neste trabalho, ou seja, obtenção da densidade

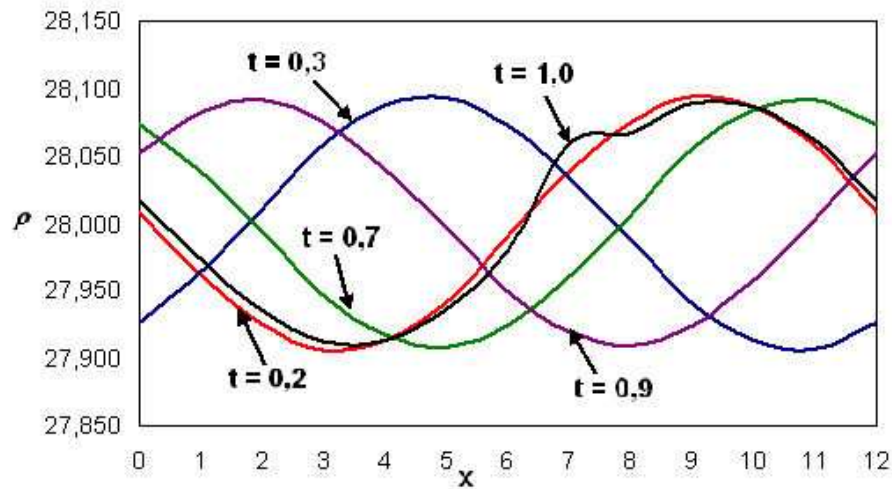


Figura 3.6: Variação de ρ pelo método das características

veicular e velocidade média nos pontos $(P_{1C_1}, P_{1C_2}, P_{2C_2}, \dots)$ da malha discretizada, facilmente consegue-se determinar os resultados para uma malha retangular através de interpolação.

A equação reduzida de Pavari Fontana foi resolvida através do método das características das distâncias especificadas para o caso particular, no qual o tráfego é homogêneo e estacionário, sendo a velocidade média desejada diretamente proporcional a velocidade atual.

Observa-se que a densidade veicular permanece praticamente constante e que a velocidade média é uma função decrescente para as sete primeiras curvas características positivas como mostra a Figura 3.7. Estes comportamentos se estendem ao longo de todas as outras curvas características positivas que compõe a malha discretizada.

O esquema do método das características utilizado mostrou-se adequado para solucionar o sistema de equações quase-lineares (3.3) e (3.4) que governam o fluxo de tráfego simulado. Em geral o esquema do método das características das distâncias especificadas não apresenta grandes dificuldades para ser implementado computacionalmente quando usado para solucionar problemas de fluxo unidimensional, como o caso do problema investigado.

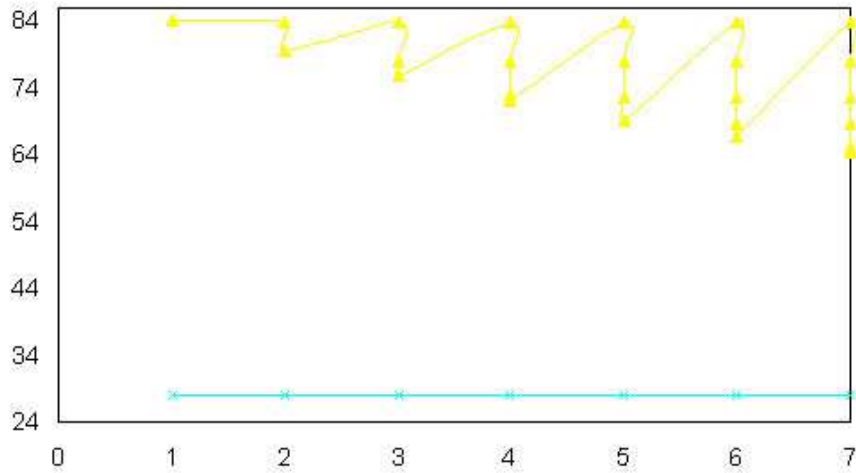


Figura 3.7: Variação da velocidade v e da densidade ρ para sete curvas características

3.5 Método de diferenças finitas explícito

Os métodos de diferenças finitas são classificados em esquemas explícitos e implícitos. Os esquemas explícitos, onde inclui-se o esquema de Lax-Wendroff (Smith, 1985), são chamados de condicionalmente estáveis, ou seja, para que sejam estáveis devem respeitar a condição de Courant-Friedrichs-Lewy (Press et al., 1992), que relaciona o intervalo de discretização no tempo em relação ao intervalo de discretização no espaço. Como consequência, necessita-se que o intervalo de tempo seja muito pequeno, independente do regime do escoamento.

Representando-se as equações (3.9) e (3.10) na forma conservativa tem-se:

$$\frac{\partial u}{\partial t} + \frac{\partial F(u)}{\partial x} = 0 \quad (3.50)$$

sendo u e F vetores colunas com 2 componentes, em função da densidade ρ , da velocidade média v e da pressão de tráfego \mathcal{P} .

$$u = \begin{pmatrix} \rho \\ \rho v \end{pmatrix}, \quad F(u) = \begin{pmatrix} \rho \\ \rho v^2 + P \end{pmatrix}, \text{ e}$$

O método de Lax-Wendroff de dois passos pode ser usado para aproximar a equação (3.50) por uma equação diferencial explícita de segunda ordem exata sendo um método no tempo de

segunda ordem que evita grandes dissipações numéricas.

Considerando-se que $u(j\Delta x, n\Delta t) = u_j^n$, definem-se valores intermediários $u_{j+\frac{1}{2}}$ para os intervalos médios do tempo $t_{j+\frac{1}{2}}$ e os pontos médios da malha $x_{j+\frac{1}{2}}$. Estes são calculados pelo esquema de Lax-Wendroff, através da equação.

$$u_{j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{2}(u_{j+1}^n + u_j^n) - \frac{\Delta t}{2\Delta x}(F_{j+1}^n - F_j^n) + \frac{\Delta t}{2}(s_i^n + s_{i+1}^n) \quad (3.51)$$

com

$$s(u) = \begin{pmatrix} q \\ \rho \frac{W-1}{\tau} v - \rho(1-p)\mathcal{P} - qv \end{pmatrix} \quad (3.52)$$

Na equação (3.51) se calcula $F_{j+\frac{1}{2}}^{n+\frac{1}{2}}$ para que os valores atualizados de u_j^{n+1} sejam calculados pela expressão adequadamente centrada:

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x}(F_{j+\frac{1}{2}}^{n+\frac{1}{2}} - F_{j-\frac{1}{2}}^{n-\frac{1}{2}}) + \frac{\Delta t}{2}(s_{j+\frac{1}{2}}^{n+\frac{1}{2}} + s_{j-\frac{1}{2}}^{n-\frac{1}{2}}) \quad (3.53)$$

3.6 Simulação numérica do método de diferenças finitas explícito (mdf)

Para completar a comparação, analisou-se dois problemas: o primeiro simula o movimento numa pista circular (programa (2) em Fortran no apêndice A) e o segundo numa pista com uma entrada lateral através de uma rua transversal (programa (3) em Delphi no apêndice A). No caso da pista circular, complementou-se a comparação com o método das características, para o caso da pista com uma entrada lateral mostrou-se uma simulação cujos resultados ainda não foram apresentados em outros trabalhos.

3.6.1 Para uma pista circular

Considera-se uma pista em que existe um tráfego homogêneo e estacionário que passa a sofrer uma pequena perturbação periódica na velocidade média (Helbing, 1996) fazendo com

que alguns veículos se movam mais rapidamente e outros mais lentamente. As condições iniciais consideradas são:

$$\rho(x, 0) = \rho_e \quad e \quad v(x, 0) = v_e \left(1 + 0,01 \sin \left(\frac{2\pi x}{L} \right) \right), \quad (3.54)$$

com $0 \leq x \leq L$ em Km. Sendo $\rho_e = 28$ veículos/km e $v_e = 84$ km/h as soluções do regime estacionário, considera-se a extensão da pista igual a $L = 12$ km e a constante de relaxação de tempo dada por $\tau = \frac{1}{120}$ h. Adota-se $\hat{\rho} = 140$ veículos/km, $V_0 = 80$ km/h e $\alpha = 100$, dados obtidos a partir de resultados experimentais.

Como a pista é circular as condições de contorno são periódicas para a densidade veicular:

$$\rho(0, t) = \rho(L, t) \text{ e para a velocidade média } v(0, t) = v(L, t).$$

Este problema foi solucionado pelo esquema numérico descrito na seção (3.5). O trecho da pista de 12 km foi subdividido em 12000 intervalos iguais a 0.001 km.

A condição de Courant-Friedrichs-Lewy (CFL)(Press et al., 1992) que relaciona o intervalo de discretização no tempo em relação ao intervalo de discretização no espaço foi verificada. Como consequência, adotou-se $\Delta t = 0,01 \times \Delta x$, um intervalo de tempo suficientemente pequeno, independente do regime do escoamento garantindo-se as condições de consistência, convergência e estabilidade do esquema numérico adotado.

Na Figura 3.8 apresenta-se a evolução da densidade veicular ρ em função do trecho da estrada $x(\text{km})$ e do tempo $t(\text{h})$. Em comparação com a Figura 3.3, observa-se que todos os tempos estão com variação uniforme. Na figura 3.9 apresenta-se a velocidade média v em função do trecho da estrada $x(\text{km})$ e do tempo $t(\text{h})$.

Velasco et al.(2005) simularam este problema, de pista circular, considerando a velocidade $V_0 = kc$, sendo $k = 1 + \frac{\rho_e(1-p)v_e\tau}{\alpha}$. Os resultados apresentados nas Figuras 3.8 e 3.9 (válidos para $V_0 = 80$ km/h) foram preparados com o propósito de realizar uma análise comparativa com os resultados de Velasco et al.(2005). Desta análise concluí-se que os resultados são semelhantes e permanecem oscilando ao longo do tempo como nos resultados de Velasco

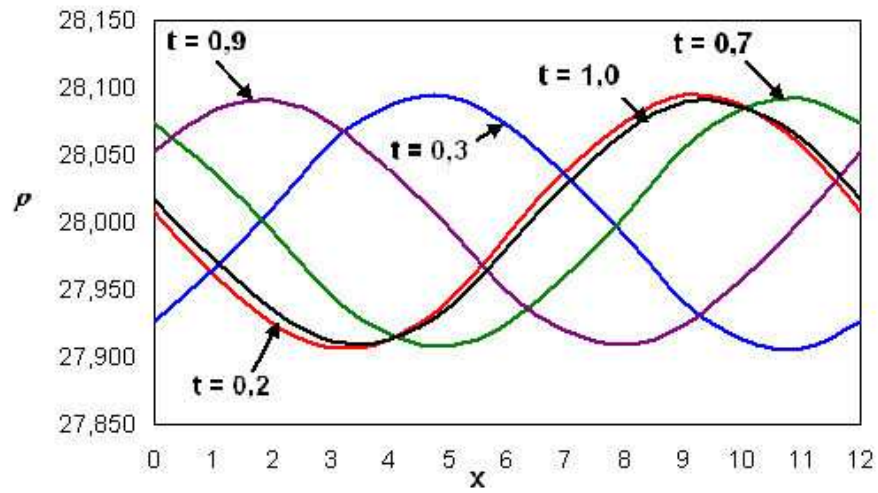


Figura 3.8: Evolução da densidade veicular ρ (veic./km) pelo mdf no apêndice A, anexo (2)

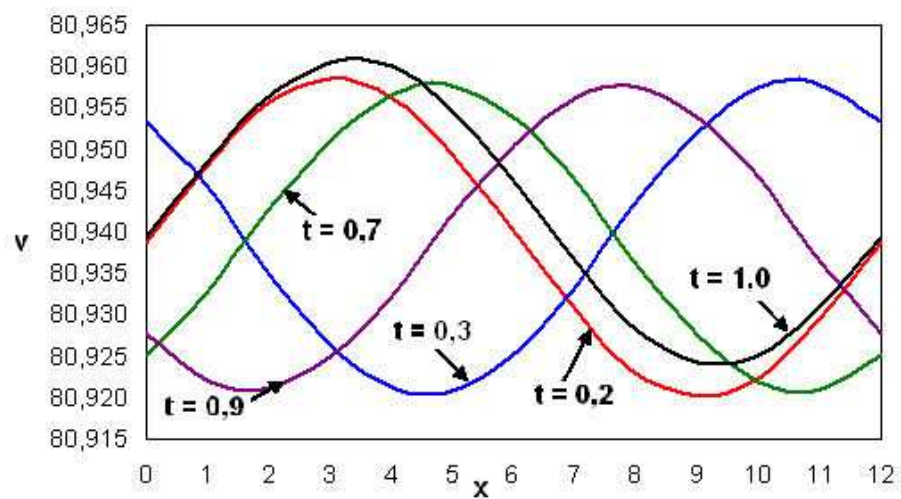


Figura 3.9: Evolução da velocidade média v (km) pelo mdf no apêndice A, anexo (2)

et al.(2005).

3.6.2 Comparação entre o método das características com o método das diferenças finitas

Para se analisar a diferença dos resultados apresentados nas Figuras (3.3, 3.8 e 3.9) referentes à densidade veicular ρ e da velocidade média v obtidos da aplicação dos métodos das características e das diferenças finitas explícito, tabulam-se o número de vezes que cada valor ocorreu. Este procedimento fornece a frequência acumulada de cada valor. A distribuição de frequência pode ser representada por um polígono de frequência acumulada.

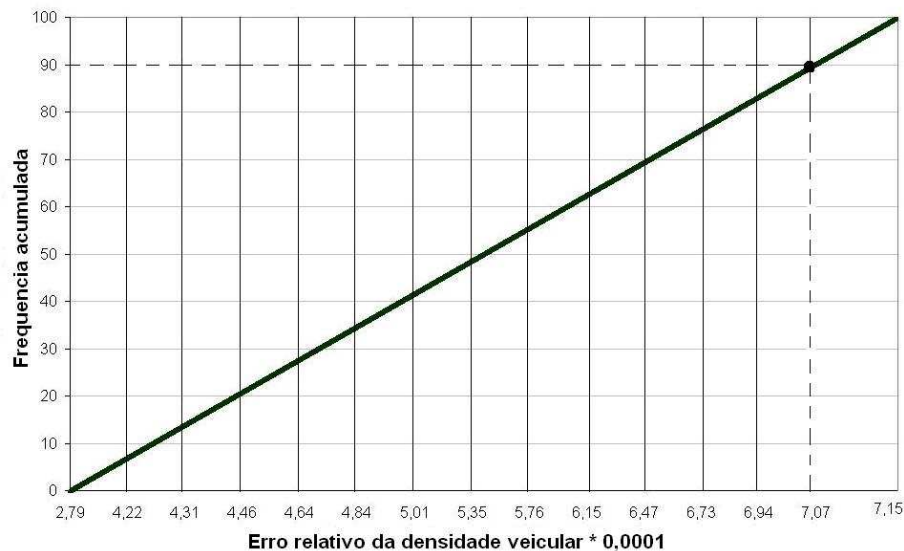


Figura 3.10: Erro Relativo vezes 10^{-4} da densidade veicular (ρ) por métodos mc e mdx x Frequência acumulada

Nas figuras 3.10 e 3.11 são apresentadas as distribuições acumuladas dos erros relativos para a densidade veicular ρ e da velocidade média v . Por exemplo, na Figura 3.10, para uma frequência acumulada de 90% tem-se que 90% dos erros relativos nos resultados são menores ou iguais a $7,07 \times 10^{-4}$, isto é, somente 10% dos resultados apresentam erros entre $7,07 \times 10^{-4}$ e $7,207 \times 10^{-4}$. Esta mesma interpretação pode ser feita para outros valores de frequência acumulada tanto para a densidade veicular quanto para a velocidade média. Estas distribuições demonstram a precisão dos resultados obtidos. Para a densidade veicular foi obtido um erro mínimo relativo de $2,788 \times 10^{-4}$ e um erro máximo relativo de $7,207 \times 10^{-4}$, e para a velocidade

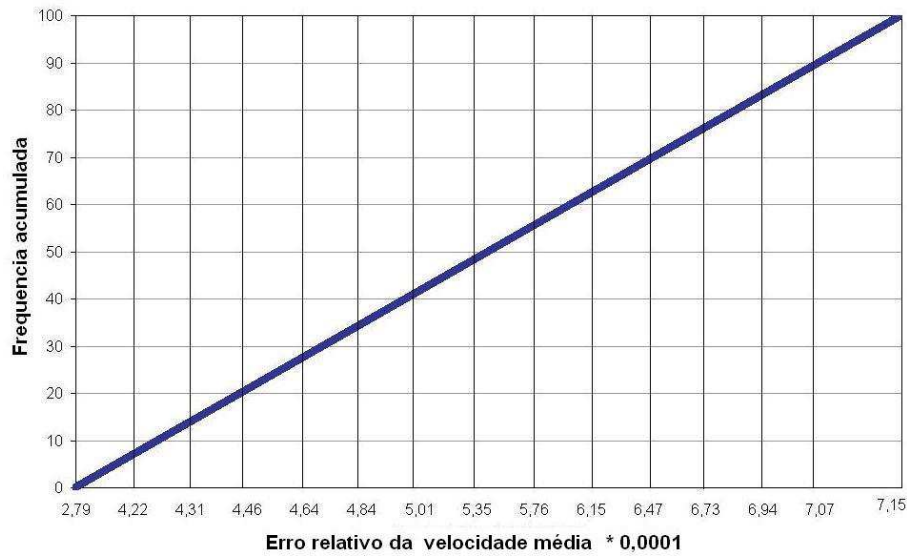


Figura 3.11: Erro Relativo vezes 10^{-4} da velocidade média (v) por mc e mdf x Frequência acumulada

média o erro varia entre $0,111 \times 10^{-4}$ e $0,591 \times 10^{-4}$.

3.6.3 Para uma pista única com uma entrada lateral

Nesta seção adota-se as mesmas condições usadas na pista circular porém sem a periodicidade nas mesmas. Então considera-se um tráfego de mão única, numa pista com extensão finita igual a $L = 12$ km, com a constante de relaxação de tempo igual a $\tau = \frac{1}{120}$ h e $\hat{\rho} = 140$ veículos/km, $\alpha = 100$ e $V_0 = 84$ km/h. As condições iniciais em $x = 0$ e $t = 0$ são dadas pela equação (3.54)

$$\rho(x, 0) = 28 \text{ veículos/km} \quad \text{e} \quad v(x, 0) = 84 \left(1 + 0,01 \sin \left(\frac{2\pi x}{L} \right) \right) \text{ km/h},$$

Analisando-se as inclinações das linhas características positivas e negativas para o problema em questão, verifica-se que têm inclinações positivas o que caracteriza o movimento como super-crítico pois neste regime de escoamento, as perturbações se projetam na direção do movimento (Abbot, 1980). Por esta razão necessita-se conhecer as condições de contorno completas em $x = 0$ escolheu-se:

$$\rho(0, t) = 28(1 + 0,1 \sin \pi t) \text{ veículos/km} \quad \text{e} \quad v(0, t) = 84 \text{ km/h} \quad (3.55)$$

Considera-se a entrada lateral localizada a 7,2 km do início pista com 6 m de largura. Como q é a densidade de carros entrando, a unidade de q é veículos por quilômetro pelo tempo. Para uma largura de pista igual a seis metros, podendo abranger 60 veículos por hora, então $q = \frac{60}{6 \cdot 10^{-3}} = 10^4$ para 1 hora de simulação.

Na figura 3.12 apresenta-se a evolução da densidade de carros entrando na estrada.

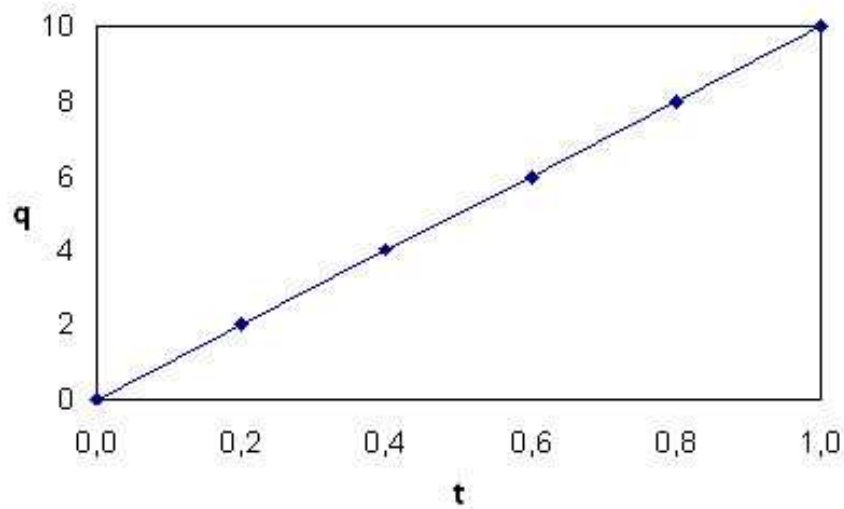


Figura 3.12: Evolução da densidade de carros entrando

A Figura 3.13 apresenta a evolução da velocidade pelo método de diferenças finitas explícitos em função do comprimento da estrada, para alguns instantes de tempo.

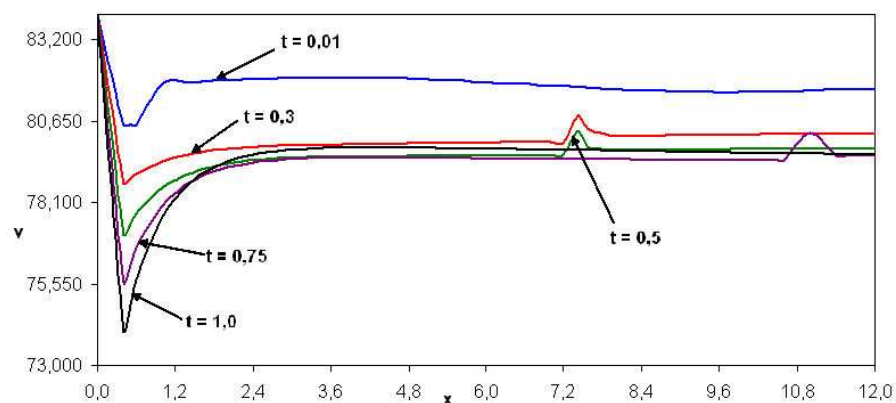


Figura 3.13: Velocidade em função da posição na estrada no apêndice A, anexo (3)

A Figura 3.14 apresenta a evolução da densidade pelo método de diferenças finitas explícito em função da posição na estrada, para alguns instantes de tempo.

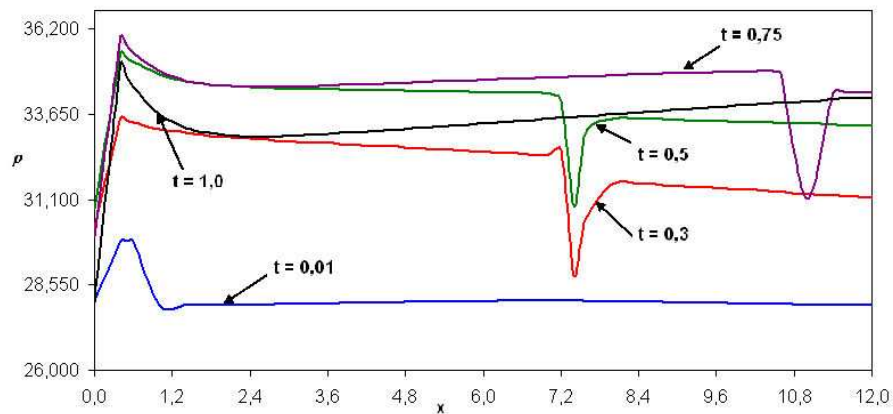


Figura 3.14: Densidade em função da posição na estrada no apêndice A, anexo (3)

3.7 Aspectos conclusivos para o método das diferenças finitas explícito

Nesta seção 3.6 soluciona-se pelo método das diferenças finitas explícito, através do esquema de Lax-Wendorff, as equações macroscópicas do fluxo de tráfego, para o caso particular em que a velocidade média prevista é considerada constante.

Os esquemas de diferenças finitas explícitos são condicionalmente estáveis. Atualmente não existem procedimentos rigorosos para determinação da estabilidade de sistemas não-lineares (Steinstrasser, 2005), porém a estabilidade pode ser investigada se os termos não-lineares forem submetidos a um processo de linearização. A condição CFL é fundamental para garantir a estabilidade e convergência do método. Nas aplicações (como neste caso), algumas vezes é necessário utilizar um intervalo de tempo menor do que o dado pela condição CFL.

Em função dos resultados obtidos para os dois problemas analisados pista circular e pista única com entrada lateral, pode-se concluir que o esquema de Lax-Wendorff é adequado para solucionar as equações macroscópicas do fluxo de tráfego, supondo-se a velocidade média prevista como constante. Em problemas nos quais as condições iniciais apresentam descontinuidades, quando as condições de contorno variam abruptamente podem ocorrer soluções oscilatórias (LeVeque, 2002) ou irregularidades em forma de serra (Ligget et al., 1975).

Para o caso da pista única com entrada lateral considerou-se uma entrada a 7,2 km do início

da mesma. Nota-se que tanto a densidade veicular quanto a velocidade média sofrem alterações neste ponto, entretanto o fator de destaque é que a densidade inicialmente sofre uma grande perturbação que com o passar do tempo se estabiliza.

Apesar disso o método de diferenças finitas explícito não consegue fazer análises qualitativas dos resultados, como por exemplo, sobre a mudança do regime do escoamento. Para isto as soluções obtidas com o método das características são usadas para verificar a precisão dos resultados determinados com o uso de outros métodos pois consegue-se facilmente identificar o regime do movimento. Este fato torna o método das características recomendado para análises quantitativas, que podem ser realizadas, mesmo sem a necessidade da obtenção de soluções numéricas (Henderson, 1966).

4 *Modelo da velocidade discreta*

O modelo das velocidades discretas para o fluxo de veículos com mesma probabilidade de acontecerem, ou seja, os veículos que trafegam ao longo da pista tendem a se moverem em grupos com mesma velocidade discreta foi proposto por Coscia et al. (2007) e Delitala et al. (2007). O trabalho de Coscia et al. supõe que a discretização das velocidades depende das condições médias locais do tráfego, enquanto que Delitala et al. supõe uma discretização em valores constantes em relação ao tempo e ao espaço, tal que globalmente o conjunto de velocidades poderá ser obtido sem a influência das condições de tráfego.

O modelo das velocidades discretas proposto por Coscia et al. (2007) supos que a variável da velocidade pode atingir um número finito n de velocidades. A discretização tratada utiliza um número fixo de velocidades com um tamanho dependendo das condições médias locais do tráfego via densidade macroscópica ρ . Em particular, o tamanho de cada passo da velocidade diminui com o aumento da densidade de uma forma que, quando o fluxo $\rho \rightarrow 1$, sendo considerado $\rho_{max} = 1$, todas as velocidades tendem a zero.

Tendo isto em mente, Coscia considerou a seguinte discretização da variável de velocidade:

$$I_v = \{v_1 = 0, \dots, v_i, \dots, v_n = v_e, \dots, v_{2n-1} = (1 + \rho)v_e\}, \quad (4.1)$$

onde v_e é a velocidade média adimensional que pode ser dada por $v_e = \exp\{-\alpha \frac{\rho}{1-\rho}\}$ ou $v_e = 1 - \rho$.

Para simplificar na equação 4.1 supõe-se que $\rho = 1$, assim $v_{2n-1} = 2v_e$ e obtem-se uma grade simétrica que será utilizada na sequência:

$$I_v = \{v_1 = 0, \dots, v_i, \dots, v_n = v_e, \dots, v_{2n-1} = 2v_e\}, v_i = \frac{i-1}{n-1}v_e(\rho). \quad (4.2)$$

A representação discreta correspondente é obtida através da ligação para cada v_i com as funções de distribuição discreta

$$f_i = f_i(t, x) : \mathbb{R}_+ \times [0, 1] \rightarrow \mathbb{R}_+, \quad i = 1, \dots, 2n-1$$

Delitala et al. (2007) consideraram o domínio $D_v = [0, 1]$ e a discretização da velocidade com a seguinte forma:

$$0 = v_1 < v_2 < v_3 < \dots < v_{n-1} < v_n = 1, \quad (4.3)$$

em que os valores de v_i são constantes em relação ao tempo e ao espaço numa grade para a velocidade com n partições sendo $v_i = \frac{i}{n}$. Isto significa que globalmente o conjunto de velocidades poderá ser obtido sem a influência das condições de tráfego.

Com base nesta hipótese a evolução no tempo e no espaço do fluxo de tráfego é representada pela função de distribuição f como uma combinação linear de n funções delta de Dirac na variável v , como segue:

$$f(t, x, v) = \sum_{i=1}^n f_i(t, x) \delta_v(v), \quad (4.4)$$

sendo $f_i, i = 1, 2, \dots, n$ o número de veículos com velocidade v_i .

As quantidades macroscópicas, tais como a densidade, o fluxo e a energia, são funções derivadas da função de distribuição f , representadas por:

(i) a densidade de veículos

$$\rho(t, x) = \sum_{i=1}^n f_i(t, x), \text{ com } \rho \in [0, 1] \quad (4.5)$$

(ii) o fluxo de veículos

$$q(t, x) = \sum_{i=1}^n v_i f_i(t, x), \text{ com } q < 1 \quad (4.6)$$

(iii) a energia

$$E(t, x) = \sum_{i=1}^n v^2 f_i(t, x), \text{ com } q < 1. \quad (4.7)$$

O modelo daa velocidadea discretaa será aplicado na escala cinética para o caso espacialmente homogêneo.

4.1 Caso espacialmente homogêneo

Nesta seção detalha-se a descrição cinética do fluxo de tráfego veicular espacialmente homogêneo ao longo de uma pista. Utilizando a equação de Boltzmann (2.24):

$$\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} + \frac{F}{m} \frac{\partial f}{\partial v} = \frac{\partial f}{\partial t} \Big|_{colisao}$$

O termo $\frac{F}{m} \frac{\partial f}{\partial v}$ pode ser desprezado pois está ligado a força externa, que neste caso não ocorre.

Para o caso espacialmente homogêneo a função de distribuição f é constante com relação a x e com isto $\frac{\partial f}{\partial x} = 0$. O termo $\frac{\partial f}{\partial t} \Big|_{colisao}$ é equivalente a $J[f, f]$ que pode ser considerado como $G_i - L_i$ (ganhos menos perdas). A equação (2.23) para a função de distribuição $f(t, x, v)$, pode ser reformulada para $f(t, v)$ e transformada para um sistema de n equações, através da aplicação do método das velocidades discretas (n é o número de velocidades discretas). Em geral para o caso espacialmente homogêneo, a equação de evolução é representada por:

$$\frac{\partial f_i}{\partial t} = G_i - L_i, \quad i = 1, 2, \dots, n \quad (4.8)$$

As equações diferenciais parciais do sistema (4.8) para as funções de distribuição $f_i(t)$, $i = 1, \dots, n$ são compostas por três termos, denominados: taxa de variação temporal $\frac{\partial f_i}{\partial t}$, G_i e L_i respectivamente os termos de ganhos e de perdas do número de veículos com velocidade v_i depois da interação com outro veículo e a diferença entre eles, $G_i - L_i$, denomina-se de termo de fontes. A estrutura matemática para este caso é dada por:

$$\frac{\partial f_i}{\partial t} = \underbrace{\sum_{h=1}^n \sum_{k=1}^n \eta_{hk}[\rho] A_{hk}^i f_h(t, x) f_k(t, x)}_{G_i} - \underbrace{f_i(t, x) \sum_{k=1}^n \eta_{hk}[\rho] f_k(t, x)}_{L_i}. \quad (4.9)$$

Este modelo é caracterizado pelas seguintes quantidades:

- Três categorias de veículos são considerados:
 - o veículo candidato (v_h): é a velocidade do veículo que após uma interação com um veículo de mesma velocidade ou velocidade diferente poderá ou não alterar a sua velocidade final.
 - o veículo campo (v_k): é a velocidade do veículo com o qual o veículo candidato interage.
 - o veículo teste (v_i): é a velocidade do veículo após a interação do veículo candidato com o veículo campo.
- $\eta_{hk}[\rho]$ é a razão de interação, o qual representa o número de interações por unidade de tempo entre os veículos com velocidade v_h e v_k . As interações são distribuídas sob um comprimento característico $\varepsilon > 0$, o qual é interpretado como a zona de visibilidade, entretanto supõe-se que as interações ocorrem com maior ou menor frequência de acordo com o comprimento desta zona de visibilidade. Espacialmente, $\eta_{hk}[\rho]$ é estimado via a quantidade $(1 - \rho)$, como segue:

$$\eta_{hk}[\rho] = \frac{c_0}{1 - \rho}, \quad (4.10)$$

sendo c_0 uma constante.

- A_{hk}^i define a densidade de probabilidade que o veículo candidato com velocidade v_h ajusta sua velocidade para v_i , depois de interagir com um veículo campo de velocidade v_k . Depende da densidade local e caracteriza-se por $A_{hk}^i[\rho] \geq 0$, $\sum_{i=1}^n A_{hk}^i[\rho] = 1$, $\forall h, k = 1, \dots, n$ para todo ρ .

4.2 Representação do sistema veículo-motorista através da partícula ativa

Esta seção trata da análise do comportamento do motorista como uma partícula ativa para o estudo do tráfego veicular em uma pista para o caso espacialmente homogêneo na escala cinética. A partícula ativa representa a qualidade do motorista, agressivo, tímido, etc., em um sistema veículo-motorista. A densidade de probabilidade total da função de distribuição é representada como função da variável discreta (nesta caso a partícula ativa), e assume-se ser constante.

Com o modelo cinético das velocidades discretas e a equação (4.9) utiliza-se as seguintes quantidades de referências para as simulações feitas:

- L é o comprimento da pista,
- V_M é a velocidade máxima e representa a velocidade de um veículo isolado movendo-se em condições de tráfego livre e
- $\bar{\rho}_{max}$ é a densidade máxima de veículos que a pista suporta.

Com base nas quantidades dimensionais acima, as seguintes variáveis adimensionais são introduzidas:

- (i) x é a posição referida do veículo a L ;
- (ii) t é o tempo normalizado representado L/V_M (utiliza-se que $x = vt$ e isola-se t);
- (iii) V é a velocidade do veículo $v = V/V_M$ é a velocidade adimensional referida a V_M ;
- (iv) $\bar{\rho}$ é a densidade de veículos e
- (v) $\rho = \bar{\rho}/\bar{\rho}_{max}$ é a densidade adimensional referida as quantidades máximas.

4.2.1 Fundamentação teórica

Nesta seção descreve-se as quantidades fenomenológicas utilizadas, ou seja, o método para determinar a distribuição do perfil dos motoristas como uma variável aleatória Y distribuída probabilisticamente. Para isto escolheu-se o modelo de Gauss. As quantidades de referência foram selecionadas a partir das seguintes considerações:

- (i) ρ é a densidade dos veículos na estrada;
- (ii) μ é a média da distribuição de Gauss (do perfil dos motoristas);
- (iii) σ é o desvio padrão da distribuição de Gauss;
- (iv) a_m é a atividade do motorista e
- (v) P é a probabilidade de cada intervalo de Y .

Considerando-se as seguintes hipóteses:

- (i) os parâmetros da distribuição de Gauss, μ e σ , são funções conhecidas da densidade ρ da ocupação dos veículos na estrada;
- (ii) as proporções de cada um dos n tipos de perfis de motoristas são conhecidas, para algum valor de ρ . Para os valores iniciais de ρ_0 , sendo $\mu_0 = \mu(\rho_0)$ e $\sigma_0 = \sigma(\rho_0)$, as proporções $P_i(\rho_0)$ de cada um dos a_m tipos de motoristas são conhecidas. Estas proporções podem ser usadas para definir o seguinte sistema de equações:

$$P_i(\rho_0) = \int_{x_{i-1}}^{x_i} g_Y(Y; \mu_0; \sigma_0) dY, \quad i = 1, \dots, a_m \quad (4.11)$$

sendo $x_0 = -\infty$ e $x_{a_m} = +\infty$, e

$$f_X(x; \mu; \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (4.12)$$

é a função densidade de probabilidade de Gauss.

O sistema de equações (4.11) pode ser usado para determinar os valores de $x_i = 1, \dots, a_{m-1}$. Para quaisquer outros valores de ρ , as proporções para cada categoria de motoristas são determinadas por:

$$P_i(\rho) = \int_{x_{i-1}}^{x_i} f_X(x; \mu(\rho); \sigma(\rho)) dx \quad (4.13)$$

4.2.2 Perfil dos motoristas

Nesta seção propõe-se um método para determinar a distribuição do perfil dos motoristas. Para ilustrar a aplicação do método proposto, consideram-se as seguintes relações:

$$\mu(\rho) = \rho, \quad \sigma = (1 - \rho^2)^{1/2}, \quad \text{para } 0 \leq \rho \leq 1. \quad (4.14)$$

Trabalha-se com a hipótese da distribuição ser normal, e que também são conhecidos os perfis de quatro motoristas ($a_m = 4$). Inicialmente para $P_1(0)$ representando o motorista calmo; $P_2(0)$ o motorista menos calmo; $P_3(0)$ o motorista agitado e $P_4(0)$ o motorista agressivo. Foram analisados cinco casos, que estão relacionados na tabela 4.2.

Tabela 4.2: Relação de casos analisados

| Caso | $P_1(0)$ | $P_2(0)$ | $P_3(0)$ | $P_4(0)$ | x_1 | x_2 | x_3 |
|------|----------|----------|----------|----------|----------|--------|--------|
| 1 | 0,25 | 0,25 | 0,25 | 0,25 | - 0,6742 | 0,0000 | 0,6742 |
| 2 | 0,30 | 0,20 | 0,20 | 0,30 | - 0,5240 | 0,0000 | 0,5240 |
| 3 | 0,50 | 0,20 | 0,15 | 0,15 | 0,0000 | 0,5240 | 1,0364 |
| 4 | 0,60 | 0,20 | 0,10 | 0,10 | 0,2529 | 0,8415 | 1,2817 |
| 5 | 0,75 | 0,10 | 0,10 | 0,05 | 0,6741 | 1,0364 | 1,6452 |

Fonte: programa (4), apêndice A, anexo (4).

Os resultados obtidos para cada um dos casos são apresentados nas Figuras 4.15 a 4.19, lembrando que cada $P_i(\rho_0) = \int_{x_{i-1}}^{x_i} f_X(x; \mu_0, \sigma_0) dx$, $i = 1, \dots, a_m$ pela equação (4.10) pode ser usado para determinar os valores de x_i , $i = 1, \dots, a_{m-1}$, pois $P_i(0) = \int_{x_{i-1}}^{x_i} f_x(x; \mu_0, \sigma_0) dx = F(x_i) - F(x_{i-1})$, com a utilização da tabela estatística (Triola, 2005).

Para a distribuição do caso 1, quando os perfis iniciais dos motoristas estão com a mesma

probabilidade ($P_1(0) = P_2(0) = P_3(0) = P_4(0) = 0,25$), (vide Figura 4.15), observa-se que com o aumento da densidade há uma alteração no comportamento dos motoristas, ao final todos estão com o perfil agressivo, ou seja, $P_1(1) = P_2(1) = P_3(1) = 0$ e $P_4(1) = 1,0$. O mesmo ocorre quando a probabilidade inicial de motoristas calmos e agressivos são iguais porém maiores que as outras duas classes, vide Figura 4.16.

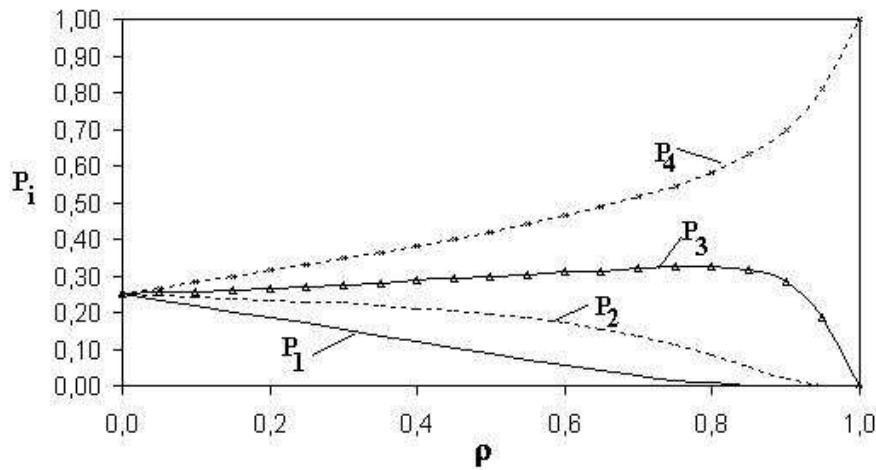


Figura 4.15: Probabilidade P versus densidade ρ (caso 1)

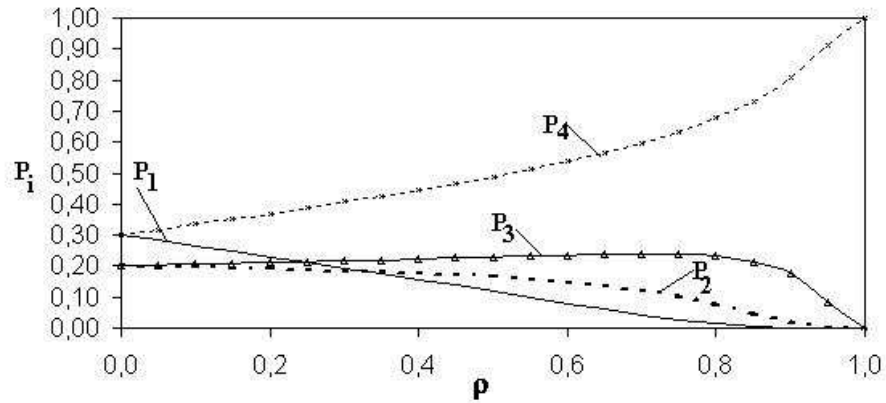


Figura 4.16: Probabilidade P versus densidade ρ (caso 2)

Os resultados na Figura 4.17 apresentam uma proporção maior de motoristas calmos $P_1(\rho)$, na proporção de 5/2 maior que os motoristas menos calmos $P_2(\rho)$ e 10/3 maior que os motoristas agitados $P_3(\rho)$ e agressivos $P_4(\rho)$. Com o crescimento da densidade, todos migram para o tipo de motorista agitado $P_3(\rho)$.

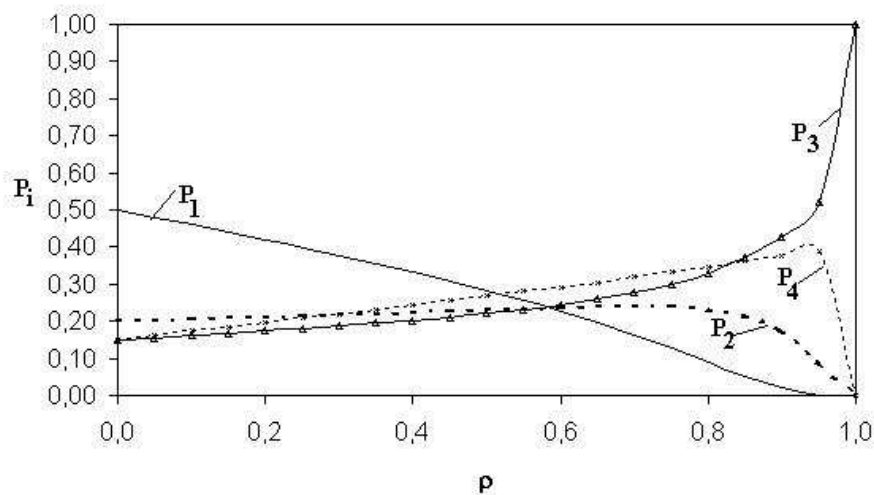


Figura 4.17: Probabilidade P versus densidade ρ (caso 3)

Tabela 4.3: Resultados para o caso 4.

| ρ | $P_1(\rho)$ | $P_2(\rho)$ | $P_3(\rho)$ | $P_4(\rho)$ |
|--------|-------------|-------------|-------------|-------------|
| 0,00 | 0,6000000 | 0,2000000 | 0,1000000 | 0,1000000 |
| 0,05 | 0,5805057 | 0,2054446 | 0,1053124 | 0,1087373 |
| 0,10 | 0,5610783 | 0,2108439 | 0,1105984 | 0,1174794 |
| 0,15 | 0,5414595 | 0,2163800 | 0,1159896 | 0,1261710 |
| 0,20 | 0,5215423 | 0,2221221 | 0,1215465 | 0,1347891 |
| 0,25 | 0,5012085 | 0,2281433 | 0,1273379 | 0,1433102 |
| 0,30 | 0,4803244 | 0,2345226 | 0,1334440 | 0,1517090 |
| 0,35 | 0,4587351 | 0,2413482 | 0,1399609 | 0,1599557 |
| 0,40 | 0,4362582 | 0,2487211 | 0,1470071 | 0,1680135 |
| 0,45 | 0,4126739 | 0,2567588 | 0,1547329 | 0,1758343 |
| 0,50 | 0,3877118 | 0,2656010 | 0,1633347 | 0,1833525 |
| 0,55 | 0,3610333 | 0,2754149 | 0,1730774 | 0,1904745 |
| 0,60 | 0,3322048 | 0,2864008 | 0,1843320 | 0,1970623 |
| 0,65 | 0,3006606 | 0,2987950 | 0,1976406 | 0,2029037 |
| 0,70 | 0,2656513 | 0,3128570 | 0,2138363 | 0,2076554 |
| 0,75 | 0,2261776 | 0,3288087 | 0,2342859 | 0,2107278 |
| 0,80 | 0,1809428 | 0,3466000 | 0,2614355 | 0,2110216 |
| 0,85 | 0,1285187 | 0,3650116 | 0,3002356 | 0,2062341 |
| 0,90 | 0,0688418 | 0,3777379 | 0,3628362 | 0,1905841 |
| 0,95 | 0,0127942 | 0,3512699 | 0,4919047 | 0,1440311 |
| 1,00 | 0,0000000 | 0,0000000 | 1,0000000 | 0,0000000 |

Fonte: Resultados do programa no apêndice A, anexo (4)

Na tabela 4.3 tem-se a proporção inicial de motoristas calmos $P_1(\rho)$ é 3 vezes maior que de motoristas menos calmos $P_2(\rho)$ e 6 vezes maior que os motoristas agitados $P_3(\rho)$ e os agressivos

$P_4(\rho)$. Com o crescimento da densidade, todos migram para o tipo de motoristas menos calmos $P_2(\rho)$.

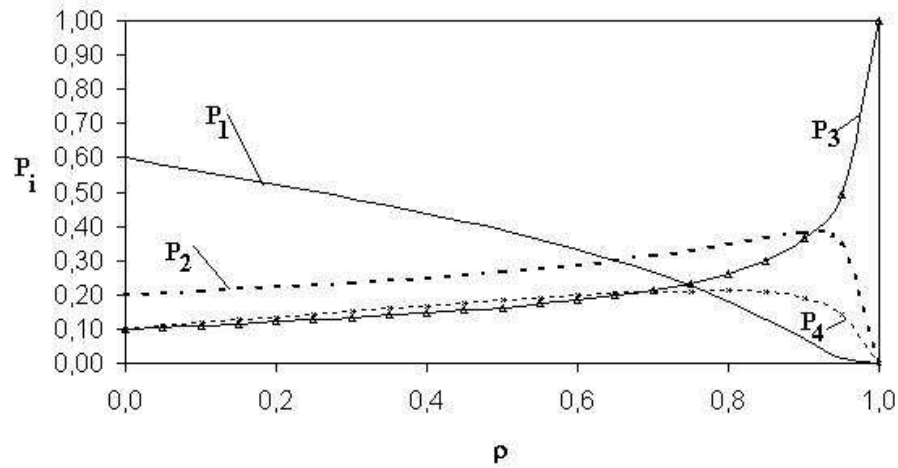


Figura 4.18: Probabilidade P versus densidade ρ (caso 4)

Os resultados para o caso 5 apresentados na Figura 4.19 apresentam uma proporção maior para os motoristas calmos $P_1(\rho)$ na proporção 15/2 maior que os motoristas menos calmos $P_2(\rho)$ e agitados $P_3(\rho)$ e 25 vezes maior que os motoristas agressivos $P_4(\rho)$. Nestas condições, com o crescimento da densidade, ao final todos migram para o tipo de motoristas menos calmos, $P_2(\rho)$.

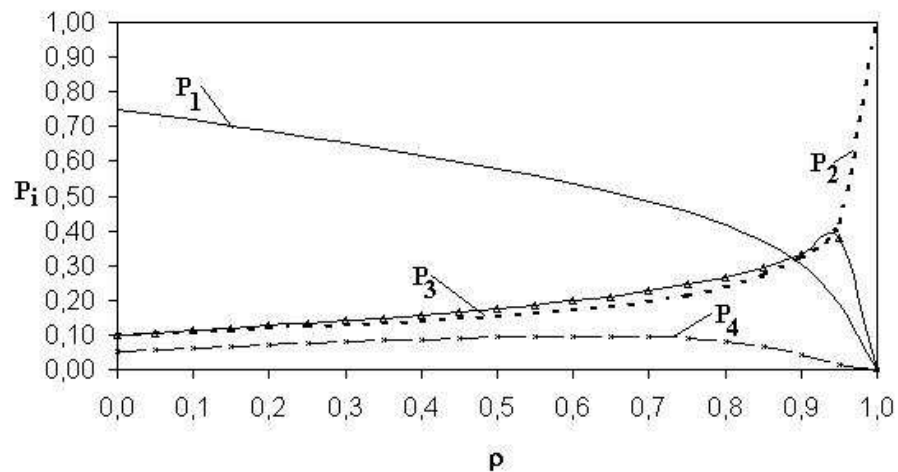


Figura 4.19: Probabilidade P versus densidade ρ (caso 5)

4.2.3 Introdução ao método dos volumes finitos

Para solucionar o problema proposto pelo sistema de equações (4.9), ou seja,

$$\frac{\partial f_i}{\partial t} = \underbrace{\sum_{h=1}^n \sum_{k=1}^n \eta_{hk}[\rho] A_{hk}^i f_h(t, x) f_k(t, x)}_{G_i} - \underbrace{f_i(t, x) \sum_{k=1}^n \eta_{hk}[\rho] f_k(t, x)}_{L_i}.$$

um problema de condição inicial com condições de contorno periódicas, utiliza-se o Método dos Volumes Finitos (MVF) (Yu et al., 1999).

A idéia básica desta técnica consiste na divisão do domínio (comprimento da pista) em um número de pequenos volumes de controle não-superpostos, procedendo a discretização das equações através de integração sobre cada volume finito, de modo a obter um sistema de equações algébricas que represente o fenômeno em análise. Essas equações algébricas são resolvidas conjuntamente com as especificações das condições iniciais e de contornos do domínio (Yu et al., 1999).

O algoritmo usado para resolver o problema analisado neste trabalho está apresentado na tabela (4.4). Este algoritmo foi preparado com um MVF de alta resolução com limitador de fluxo numérico "superbee" de Roe, conforme descrito em (Leveque, 1990); existem outros dois conhecidos como minmod e o van Leer. Em métodos numéricos precisa-se garantir a estabilidade para resolver equações diferenciais parciais hiperbólicas, que é o tipo de equação que resolveu-se neste capítulo/, e uma maneira é usar a região de TVD (em inglês, total variation diminishing) introduzida por Ami Harten (1983). Dos três limitadores acima, o que melhor cobre a região de TVD é o superbee, por isso ele é escolhido. Estes recursos conferem uma maior precisão nos resultados em relação à precisão obtida com os métodos de segunda ordem.

Chamando C_{fl} a condição de Courant, Friedrichs e Lewy tem-se:

No algoritmo apresentado na tabela 4.4, os índices da função de distribuição $f_{i,j}^k$ com $k = 1, 2, 3, \dots$, $i = 1, \dots, n$ e $j = 0, \dots, m$; k são os valores do instante de tempo; i representa a velocidade discreta v_i e j a posição discretizada na pista. Fixando-se m (número de intervalos de discretização do comprimento da pista), define-se $dx = L/m$, sendo L o comprimento da pista.

Tabela 4.4: Algoritmo do Método do Volumes Finitos

| | |
|--|--|
| (i) Definições | $\lambda = \Delta t / \Delta x < C_{fl}$ $f_i(t, x) = f_i(k\Delta t, j\Delta x) \equiv f_{i,j}^k$ $i = 1, \dots, n; j = -1, \dots, m+1$ $k = 0, 1, 2, \dots \quad m = \frac{L}{\Delta x}$ |
| (ii) Condições iniciais | $f_{i,j}^0$ devem ser conhecidas para $i = 1, \dots, n$ e $j = 1, \dots, m$ |
| (iii) Condições de contorno periódicas | $f_{i,-1}^k = f_{i,m-1}^k, f_{i,0}^k = f_{i,m}^k$ e $f_{i,m+1}^k = f_{i,1}^k$ |
| (iv) Solução pelo método dos volumes finitos | $f_{i,j}^{k+1} = f_{i,j}^k - \lambda \Phi_{i,j} + \Delta t S_{i,j}$ para $i = 1, \dots, n$ e $j = 1, \dots, m$ $\Phi_{i,j}$ representa o transporte convectivo $\Phi = v_i \left\{ d_1 + \frac{1-\lambda v_i}{2} \left[\frac{d_1}{\psi(d_2/d_1)} - \frac{d_2}{\psi(d_3/d_2)} \right] \right\}$ $d_1 = f_{i,j+1}^k - f_{i,j}^k$ $d_2 = f_{i,j}^k - f_{i,j-1}^k$ $d_3 = f_{i,j-1}^k - f_{i,j-2}^k$ $\psi(\theta) = \max[0, \min(1, 2\theta), \min(2, \theta)]$ $S_{i,j}$ representa a diferença entre os termos de ganhos e perdas |

Fonte: Algoritmo utilizado no apêndice A, anexo (4).

O valor de $\lambda = \frac{dt}{dx}$ deve ser determinado de forma que seja respeitada a condição de Courant, Friedrichs e Lewy (CFL) (Courant et. al, 1967), para garantir estabilidade nas soluções. Deve-se respeitar a condição $\lambda < CFL$, sendo CFL um valor inversamente proporcional a velocidade máxima do tráfego veicular. O termo $S_{i,j}$ chamado ”fontes” do algoritmo da tabela 4.4 tem por objetivo determinar o termo à direita da equação (4.9). Com a função $\Phi_{i,j}$ estima-se o termo do transporte convectivo, sendo que a função ψ é usada para calcular o limitador de alta resolução ”superbee” de Roe. O procedimento MVF é usado para solucionar o problema de valor inicial com condições de contorno periódicas. A expressão usada para obter as soluções para o próximo instante de tempo, pelo MVF implementado, são semelhantes às usadas em esquemas de diferenças finitas explícitos.

4.2.4 Resultados da simulação do motorista

Foram analisados computacionalmente dois casos usando o MVF: o primeiro é o caso 1 em que as proporções são todas iguais, neste caso $P_1(0) = P_2(0) = P_3(0) = P_4(0) = 0,25$ e o

segundo caso é o caso 4 em que a proporção $P_1(0)$ dos motoristas calmos é maior que as outras e a proporção é $P_1(0) = 0,60$; $P_2(0) = 0,20$; $P_3(0) = P_4(0) = 0,10$.

Considera-se as condições da pista como ruim ($\alpha = 0,3$), média ($\alpha = 0,6$) ou ótima ($\alpha = 1,0$). O programa (4) é iniciado com condições que permitem escolher a distribuição do perfil dos motoristas (quando seleciona-se uma determinada variável $g=2$).

Os resultados encontrados são apresentados nas tabelas 4.5 e 4.6, em que a primeira coluna representa a densidade, a segunda, terceira e quarta colunas representam o fluxo médio q e as quinta, sexta e sétima colunas representam a velocidade média v para cada uma das qualidades de pista. Os menores valores do fluxo e da velocidade média encontram-se nas pistas de qualidade ruim ($\alpha = 0,3$).

Tabela 4.5: Resultado da simulação para o caso 1.

| ρ | $q, \alpha=0,3$ | $q, \alpha=0,6$ | $q, \alpha=1,0$ | $v, \alpha=0,3$ | $v, \alpha=0,6$ | $v, \alpha=1,0$ |
|--------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| 0,00 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 |
| 0,20 | 0,00492637 | 0,01286354 | 0,04030770 | 0,02463186 | 0,06431771 | 0,20153849 |
| 0,40 | 0,00479228 | 0,01066380 | 0,02090949 | 0,01198070 | 0,02665950 | 0,05227374 |
| 0,60 | 0,00279999 | 0,00582922 | 0,01021051 | 0,00466665 | 0,00971536 | 0,01701751 |
| 0,80 | 0,00076171 | 0,00153906 | 0,00259697 | 0,00095214 | 0,00192383 | 0,00324622 |
| 1,00 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 |

Fonte: Resultado do programa(4) para simular o caso 1.

Tabela 4.6: Resultado da simulação para o caso 4.

| ρ | $q, \alpha=0,3$ | $q, \alpha=0,6$ | $q, \alpha=1,0$ | $v, \alpha=0,3$ | $v, \alpha=0,6$ | $v, \alpha=1,0$ |
|--------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| 0,00 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 |
| 0,20 | 0,00492637 | 0,01286356 | 0,04030773 | 0,02463187 | 0,06431779 | 0,20153867 |
| 0,40 | 0,00479228 | 0,01066378 | 0,02090936 | 0,01198071 | 0,02665946 | 0,05227340 |
| 0,60 | 0,00279999 | 0,00582921 | 0,01021050 | 0,00466665 | 0,00971534 | 0,01701750 |
| 0,80 | 0,00076171 | 0,00153906 | 0,00259697 | 0,00095214 | 0,00192383 | 0,00324621 |
| 1,00 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 | 0,00000000 |

Fonte: Resultado do programa(4) para simular o caso 4.

Fonte: Apêndice A no anexo (4).

Os resultados descritos nas tabelas 4.5 e 4.6 são muito semelhantes, a diferença quando ocorre está na oitava casa depois da vírgula e por isto apresenta-se apenas uma curva (vide Figura 4.20 e 4.21) referente a tabela 4.5 sendo v_1 , v_2 e v_3 , respectivamente, as velocidades

para $\alpha = 0,3; \alpha = 0,6$ e $\alpha = 1,0$; o mesmo ocorre para q_1, q_2 e q_3 , que são os fluxos para $\alpha = 0,3; \alpha = 0,6$ e $\alpha = 1,0$, respectivamente.

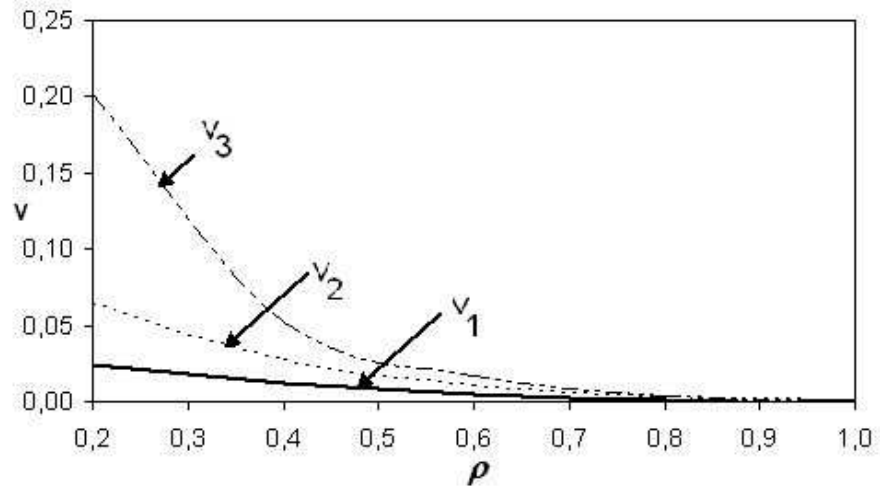


Figura 4.20: Resultado da velocidade para simular o caso 1

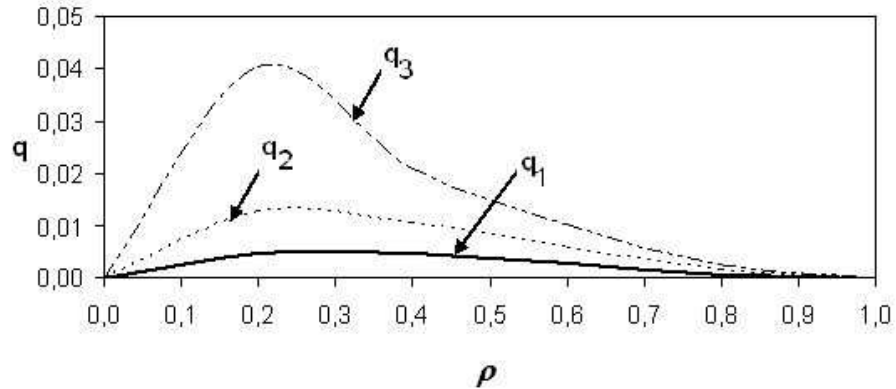


Figura 4.21: Resultado da densidade para simular o caso 1

4.2.5 Conclusões do perfil do motorista

Para tratar a análise do perfil do motorista como uma partícula ativa, que representa a qualidade do motorista (calmo, agressivo, etc.) desenvolveu-se o modelo cinético das velocidades discretas em um sistema veículo-motorista. Observa-se na seção 4.2.2 que o perfil dos motoristas altera-se dependendo das condições iniciais.

Conclui-se que a qualidade da estrada influencia fortemente o fluxo e a velocidade média (Prado,2009), enquanto que quanto ao comportamento do motorista, observa-se que ao se ter um número de motoristas agressivos maior do que um número de motoristas calmos ou agitados, o perfil é alterado para agressivo conforme a densidade cresce e o fluxo diminui. Influências deste comportamento nas condições de escoamento não foram observadas.

4.3 Análise dos efeitos de condições iniciais aleatórias nos resultados das simulações

Neste seção apresenta-se a pesquisa na análise dos efeitos das condições iniciais para o problema de fluxo de tráfego em uma pista para o caso espacialmente homogêneo na escala cinética estabelecidas por diferentes métodos de geração de números pseudo-aleatórios uniformes, sobre os resultados obtidos. Para tanto foram investigados 18 algoritmos para geração de números pseudo-aleatórios uniformes, colocados no apêndice A, anexo (5). Estes algoritmos são fundamentados nas seguintes técnicas: (i) gerador congruente multiplicativo (MC); (ii) gerador por deslocamento de registro (SR); (iii) gerador com defasagens de Fibonacci (LF); (iv) aleatorização por embaralhamento; (v) geradores combinados; e (vi) gerador RANLUX (CERN,2002). As propriedades estatísticas dos números pseudo-aleatórios devem coincidir com as propriedades estatísticas dos números verdadeiramente aleatórios (independentes e igualmente prováveis). Os números pseudo-aleatórios não são randômicos na definição do termo, porque são completamente determinados a partir de valores iniciais, têm precisão limitada e fazem parte de seqüências com período finito. Quando os números pseudo-aleatórios são aprovados por testes estatísticos, podem ser tratados como verdadeiros números aleatórios. Na literatura sobre a geração de números pseudo-aleatórios são apresentadas listas extensas de testes estatísticos. Os geradores foram analisados através de dois testes: (i) de freqüência, aplicado para verificar a uniformidade da distribuição de sucessivos conjuntos de N números num espaço com N dimensões; e (ii) da contagem do total de números pseudo-aleatórios que não são distribuídos em m células.

4.3.1 Números Pseudo-Randômicos

O termo número aleatório (ou randômico) é usado para especificar números aleatórios uniformes (tipo gaussiano). Os métodos de geração de números pseudo-randômicos tem por objetivo produzir seqüências de números:

$$u_0, u_1, u_2, \dots, \quad (4.15)$$

sendo que cada u_n , deve estar incluído no intervalo $[0, 1)$, deve ter a mesma probabilidade de ocorrência, e deve ser independente em relação aos demais valores u_k , para $n \neq k$. As técnicas computacionais mais comuns produzem seqüências de números inteiros distribuídos uniformemente:

$$x_0, x_1, x_2, \dots, \quad (4.16)$$

com $0 \leq x_n \leq B$, onde B é um valor limite. Existe um menor número positivo P , chamado de período, para o qual a relação $x_{n+P} = x_n$ ocorre para todo n . A seqüência de números uniformes pertencente ao intervalo $[0, 1)$ é determinada por:

$$x_0/B, x_1/B, \dots, x_{P-1}/B. \quad (4.17)$$

Esta seqüência não pode ser uniformemente distribuída no intervalo $[0, 1)$ porque existe somente um número finito de elementos. Quando P e B são suficientemente grandes, a seqüência pode ser considerada uniforme para todos os efeitos práticos. O valor 0 pode ocorrer, na seqüência normalizada, desde que ocorra na seqüência de x_n . Teoricamente, o valor 1 não pode ocorrer. Dependendo do arredondamento causado pela representação do sistema de ponto flutuante pode existir computacionalmente o número 1 (Kaviski, 2001).

Os geradores de números pseudo-aleatórios devem ser inicializados por um número inicial x_0 . Dependendo da técnica de geração empregada existem critérios específicos para a definição do valor inicial. Muitos sistemas computacionais empregam construções para o valor inicial

baseado no sistema de data e hora. Por exemplo (Anderson, 1990):

$$x_0 = ano - 200 + 100[(mes - 1) + 12[(dia - 1) + 31[hora + 24 \times (min + 60 \times seg)]]], \quad (4.18)$$

que satisfaz $0 \leq x_0 \leq 3214079999 \sim 2^{3158176}$.

As técnicas comumente empregadas para produção de seqüências de números pseudo-aleatórios são as seguintes (Anderson, 1990): (i) Gerador congruente multiplicativo (Multiplicative Congruential - MC); (ii) Gerador por deslocamento de registro (Shift Register - SR); (iii) Gerador com defasagens de Fibonacci (Lagged-Fibonacci - LF); (iv) Aleatorização por embaralhamento; (v) Geradores combinados; (vi) Gerador RANLUX (LUXury RANdom numbers); e (vii) Outros geradores de números pseudo-aleatórios.

4.3.1.1 Gerador congruente multiplicativo (MC)

Dos geradores em uso o método Multiplicative Congruential(MC) é o mais antigo. Foi proposto por D. H. Lehmer em 1948 (Knuth, 1981) e a seqüência de números pseudo-aleatórios é determinada por:

$$x_{n+1} = (ax_n + c) \bmod m, n \geq 0 \quad (4.19)$$

A relação $a = b \bmod r$ é definida por a ser congruo a b , módulo r , se, e somente se r divide $(b - a)$ neste caso m é chamado módulo $m(m > 0)$, a é o multiplicador ($0 < a < m$), c é um incremento ($0 \leq c < m$), e x_0 é um valor inicial ($0 \leq x_0 < m$). A seqüência correspondente de números no intervalo $[0, 1)$ é determinada por x_m/m . O maior período possível é m . Esta seqüência pode ser representada por $MC[a, c, m; x_0]$.

Em geral consideram-se três classes de geradores MC que normalmente são usados:

- $MC[a, c, 2^N; x_0]$;
- $MC[a, 0, 2^N; x_0]$ e

- $MC[a, 0, m = \text{primo}; x_0]$.

Na classe de geradores $MC[a, c, 2^N; x_0]$ o máximo período possível pode ser obtido quando c é um número ímpar, e $a - 1$ é um múltiplo de 4 (Knuth, 1981). O número 0 é um dos membros da seqüência. Este gerador é facilmente implementado em computadores binários usando-se informações sobre a representação de dados (Kaviski, 2001). Um exemplo desta classe de gerador tem os seguintes parâmetros (Knuth, 1981): $MC[5^{15}, 7261067085, 2^{35}; x_0]$. Computacionalmente este gerador foi implementado em Fortran numa rotina denominada (1)rkuth. Os geradores com parâmetros: $MC[31623, 2178281829, 2^{48}; x_0]$ servem para definir as rotinas (2)rcdcx1, (3)rcdcx2 e (4)rcdcx3 que são usadas em computadores CDC-6500 (Phillips et al., 1975). Este gerador não tem a propriedade $a - 1$ múltiplo de 4, de forma que não se consegue obter a seqüência de máximo período.

Na classe de geradores $MC[a, 0, 2^N; x_0]$ não consegue-se obter o máximo período possível igual a m . O maior período possível é 2^{N-2} , que pode ser obtido quando o parâmetro a é da forma $8k + 3$ ou $8k + 5$ e o valor inicial é ímpar (Knuth, 1981). Um exemplo desta classe de gerador tem os parâmetros: $MC[44485709377909, 0, 2^{48}; x_0]$, usado em computadores CRAY-X-MP (Anderson, 1990) e em Fortran este gerador foi implementado na rotina (5)rcrayl. As rotinas (6)rcdcl1, (7)rcdcl2, (8)rcdcl3 e (9)rcdcl4, são exemplos de implementações para o gerador com parâmetros: $MC[31623, 0, 2^{48}; x_0]$, usado em computadores CDC-6500 (Phillips et al., 1975). Este gerador não tem a propriedade descrita para o parâmetro a , de forma que não consegue-se obter a seqüência de melhor período. Um exemplo clássico desta classe de geradores com parâmetros $MC[65539, 0, 2^{31}; x_0]$ é o da sub-rotina randu (I.B.M., 1970); (Compaq, 1999). Implementadas em Fortran nas rotinas com os nomes: (10)randu1 e (11)randu2.

Na classe de geradores $MC[a, 0, m = \text{primo}; x_0]$ não consegue-se obter o máximo período possível igual a p , porque o número 0 não pode fazer parte da seqüência. O maior período possível é $p - 1$, que pode ser obtido quando o parâmetro a é um elemento primitivo módulo m (Knuth, 1981) e $x_0 \neq 0$. O mais popular gerador, incluído nesta classe, foi proposto por Lewis, Goodman e Miller em (1969): $MC[16807, 0, 2^{31} - 1; x_0]$. As implementações em Fortran foram realizadas nas rotinas (12)rlgm1 e (13)rlgm2. Outro exemplo, é o gerador proposto por Lehmer

em 1951 (Hutchinson, 2004): . No Fortran da Compaq é usado o gerador $MC[40014, 0, 2^{31} - 85; x_0]$ (Compaq, 1999), através da sub-rotina (14)random-number.

Um outro exemplo é o do gerador $MC[13967501, 0, 2^{32} - 1; x_0]$, que encontra-se disponível no Fortran Lahey (Lahey,1998), através da sub-rotina (14)random number. Neste caso o parâmetro m não é um número primo. O parâmetro a é da forma $8k + 5$. Nesta classe temos 14 tipos de geradores e escolheu-se a rknuth.

4.3.1.2 Gerador por deslocamento de registro (SR)

As variáveis consideradas neste tipo de gerador são números inteiros representados por N bits. Um gerador por deslocamento de registro, ou Shift Register, é representado por (Fushimi et al.,1983) e (Anderson, 1990): $SR[i, j, N; x_0]$. Sendo $N > 1, 0 < i, j < N$, e $0 < x_0 < 2^N$. A sequência é definida por:

$$x_{n+1} = t \oplus (tSHLj), \quad t = x_n \oplus (x_nSHRi). \quad (4.20)$$

sendo \oplus o operador OU exclusivo (compara 2 bits e retorna o resultado 1 somente se um dos bits é igual a 0 e o outro igual a 1), $SHRi$ é o operador de deslocamento para a direita de i bits e $SHLj$ é o operador de deslocamento para a esquerda de j bits.

O máximo período que pode ser obtido por este tipo de gerador é $2^N - 1$, sendo que o número 0 não pode pertencer a sequência. Para $N = 31$, os seguintes geradores, propostos por Marsaglia em 1983, apresentam máximo período (Anderson, 1990):

$$SR[3, 28, 31; x_0], SR[6, 25, 31; x_0], SR[13, 18, 31; x_0]. \quad (4.21)$$

Quando $N = 32$, não consegue-se obter o máximo período. Por exemplo, para o gerador com parâmetros: $SR[15, 17, 32; x_0]$, obtêm-se 99.95% do período total. Este gerador, proposto por Marsaglia em 1983 (Anderson, 1990), é um dos mais populares, e também, mais facilmente implementado em linguagem de alto nível.

Os geradores SR são mais facilmente implementados em linguagens Assembler, mas conhecendo-se a representação interna de dados, também podem ser implementados em linguagens de alto nível (Kaviski, 2001). Na rotina (15)rdesrg apresenta-se uma implementação do gerador $SR[15, 17, 32; x_0]$.

4.3.1.3 Gerador com defasagens de Fibonacci (LF)

Esta é uma classe de geradores que apresentam as vantagens de serem rápidos e por permitirem obter longas seqüências de números pseudo-aleatórios. São considerados por Knuth (1981) como a melhor fonte para obtenção de números pseudo-aleatórios para atingirem propósitos práticos. Geralmente não são inteiramente aceitos porque os fundamentos teóricos não são sólidos como os existentes para os outros geradores. Teme-se que nas seqüências geradas possam ocorrer emboscadas com correlações que ainda não foram descobertas. Newman e Barkema (1999) tem usado extensivamente este gerador, obtendo excelentes resultados com a solução de problemas que envolvem longas seqüências de números pseudo-aleatórios. Para proteger-se da possibilidade da ocorrência de séries correlacionadas, estes autores recomendam o uso deste gerador associado com outros geradores.

A forma geral deste tipo de gerador é representada por: $LF[r, s, m, op; x_0, \dots, x_{r-1}]$, sendo s as defasagens ($0 < s < r$), m é a base, op é um operador binário, e x_0, \dots, x_{r-1} é uma seqüência de r valores iniciais. Para $n \geq r$, a seqüência é definida por:

$$x_n = x_{n-r} .op x_{n-s}. \quad (4.22)$$

Os operadores op geralmente são os seguintes: $+$ adição e mod m ; $-$ subtração e mod m ; $*$ multiplicação e mod m ; e $\oplus OR$ exclusivo se m é uma potência de 2. Alguns exemplos de geradores propostos por Marsaglia em 1985 (Anderson, 1990), são os seguintes:

- $LF[17, 5, 2^N, \pm; x_0, \dots, x_{16}]$ com período $(2^{17} - 1)2^{N-1}$, se pelo menos um membro da seqüência inicial é ímpar;
- $LF[17, 5, 2^N, *; x_0, \dots, x_{16}]$ com período $(2^{17} - 1)2^{N-3}$, se todos os membros da seqüência

inicial são ímpares; e

- $LF[17, 5, 2^N, *; x_0, \oplus, x_{16}]$ com período $(2^{17} - 1)2^{N-3}$. O período deste último gerador independe do tamanho do número de bits da base, podendo ser muito pequeno.

Alguns geradores podem possuir períodos extremamente longos, como por exemplo, o gerador com parâmetros: $LF[607, 273, 2^{31}, -; x_0, \dots, x_{606}]$ com período $(2^{607} - 1)2^{30} \sim 10^{191}$, se pelo menos um dos membros da seqüência inicial é ímpar.

O gerador $LF[55, 31, 10^9, -; x_0, \dots, x_{54}]$ foi implementado em Fortran com a rotina (16)ranmpb. O algoritmo foi proposto por Knuth (1981) e escrito em linguagem de alto nível por Press e outros (1992). Foi adotado por Bird (1994), para utilização em aplicações com o método DSMC.

Para determinação da seqüência inicial geralmente são usados geradores simples, como por exemplo o método congruente multiplicativo. Existem demonstrações que garantem que os valores iniciais não são gerados novamente pelo gerador LF (Anderson, 1990). Marsaglia em 1985 (Anderson, 1990), recomenda a construção da seqüência inicial bit por bit usando o gerador: $LF[3, 1, 32707, -; x_0, x_1, x_2]$. Os três valores iniciais podem ser determinados através do inicializador descrito pela expressão (4.18).

4.3.1.4 Geradores combinados

Consideram-se duas seqüências de números aleatórios $\{x_n\}$ e $\{y_n\}$, distribuídos nos intervalos $[0, B_x]$ e $[0, B_y]$, respectivamente. Se $m \geq \max(B_x, B_y)$, então pode-se definir a seguinte combinação entre as seqüências (Anderson, 1990):

$$z_n = (x_n \pm y_n) \bmod m. \quad (4.23)$$

Os números uniformes no intervalo $[0, 1)$ são obtidos fazendo-se z_n/m . Sendo que, somente usa-se uma das operações $+$ ou $-$, não ambas. Se as seqüências $\{u_n\}$ e $\{v_n\}$ são uniformes no intervalo $[0, 1)$, a combinação entre as seqüências pode ser definida por:

$$w_n = (u_n \pm v_n) \bmod 1 \quad (4.24)$$

Da mesma forma, mais de duas seqüências também podem ser combinadas. Alguns exemplos de geradores combinados são os seguintes:

$$\left\{ \left(\frac{MC[171, 0, 30269; x_0]}{30269} \right) + \left(\frac{MC[172, 0, 30307; y_0]}{30307} \right) + \left(\frac{MC[170, 0, 30323; z_0]}{30323} \right) \right\} \bmod 1, \quad (4.25)$$

proposto por (Wichmann et al., 1982):

$$\{LF[2, 1, 2^{32}, *, x_0, x_1] - LF[3, 1, 2^{32} - 5, -, y_0, y_1, y_2]\} \bmod 2^{32}, \quad (4.26)$$

proposto por Marsaglia em 1985 (Anderson, 1990):

$$MC[69069, 0, 2^{32}; x_0] \oplus SR[15, 17, 32; y_0], \quad (4.27)$$

proposto por Marsaglia em 1983 (Anderson, 1990), com forma diferente dos anteriores. Recomenda-se o uso do seguinte gerador combinado:

$$\{(MC[1364, 0, 1500419; x_0]/1500419) + (MC[1528, 0, 1400159; y_0]/1400159)\} \bmod 1 \quad (4.28)$$

Este gerador foi implementado em Fortran com o nome de (17)rcomb.

4.3.1.5 Gerador RANLUX

Marsaglia e Zaman (Shchur et al., 1998) definiram uma relação recursiva envolvendo três números positivos inteiros: b chamado de base, r e s chamados de defasagens ($r > s$). Este algoritmo é conhecido como gerador SWC (Subtract-With-Carry). A qualidade deste gerador foi estabelecida através de resultados matemáticos obtidos em dinâmica de sistemas caóticos, testes espectrais e por meio da aplicação de um grande número de testes empíricos. Conhecendo-se os

primeiros r números pseudo-aleatórios x_0, x_1, \dots, x_{r-1} e o bit de passagem c_{r-1} (carry bit), o n -ésimo número pseudo-aleatório é determinado calculando-se inicialmente a seguinte diferença:

$$\Delta_n = x_{n-s} - x_{n-r} - c_{n-1}, \quad (4.29)$$

$$x_n = \Delta_n, \quad cn = 0 \text{ se } \Delta_n \geq 0,$$

$$x_n = \Delta_n + b, \quad cn = 1 \text{ se } \Delta_n < 0 \quad (4.30)$$

O máximo período possível deste gerador é $m = b^r - b^s + 1$, que pode ser atingido quando m é primo, e quando b e m são primos relativos. Os parâmetros escolhidos por Marsaglia e Zaman (Lüscher, 1994), são os seguintes: $b = 2^{24}$, $r = 24$ e $s = 10$.

Para corrigir algumas deficiências deste algoritmo, que falha em alguns testes de correlação, Lüscher (1994) propôs descartar alguns dos números pseudo-aleatórios produzidos pelo gerador SWC, usando somente os números que restarem. Este gerador é conhecido com o nome de RANLUX (LUXury RANdom numbers). O gerador produz p números pseudo-aleatórios, destes, usam-se os primeiros 24 e descartam-se os próximos $p - 24$, e assim sucessivamente. Os valores $p = 48, 97, 223$ e 389 , definem os níveis de superfluidade 1, 2, 3 e 4, respectivamente. Através de análises teóricas foi demonstrado que o nível 3 apresenta as menores chances de se observarem correlações nas séries de números pseudo-aleatórios gerados. Quando $p = 24$, geralmente chamado de nível 0, são obtidos os resultados equivalentes ao gerador SWC. O gerador RANLUX tem sido usado em aplicações desenvolvidas no CERN - Conseil Européen pour la Recherche Nucléaire (2002). Este gerador foi implementado em Fortran com o nome de (18) ranlux.

4.3.2 Algoritmo de Metropolis

Para simular exatamente um sistema físico complexo como um gás ou o conjunto de íons paramagnéticos em interação necessita-se calcular o movimento de aproximadamente 10^{23} partículas. Para cada uma destas partículas deve-se considerar um determinado número de estados (níveis de energia, posições, orientações do momento magnético, etc.), resultando num número de configurações do sistema muito grande. O método de Monte Carlo permite simular adequadamente estes sistemas através de amostras sem a necessidade de reproduzir todas as configurações do sistema.

Quando a probabilidade da geração de uma dada configuração é considerada como a mesma para todas as configurações do espaço de fase o método é chamado de método amostral simples. Esta forma de cálculo pode demandar muito tempo computacional e ser muito ineficiente porque muitas das configurações geradas podem apresentar um peso inexpressivo no cálculo que esta sendo determinado. Este problema pode ser superado implementando-se o algoritmo de Metropolis (Metropolis et. al., 1953), usado para produzir variáveis aleatórias com uma determinada distribuição de probabilidades com forma arbitrária.

Formula-se o algoritmo de Metropolis, supondo-se que pretende-se gerar um conjunto de pontos num espaço de variáveis X distribuídas com função densidade de probabilidade $f_X(X)$. Com o algoritmo de Metropolis gera-se uma seqüência de pontos, X_0, X_1, \dots , representando um caminho aleatório movendo-se através do espaço X . As regras pelas quais o caminho aleatório é realizado no espaço configurado são as seguintes:

- Considera-se que o caminho aleatório encontra-se no ponto X_n .
- Para gerar o ponto X_{n+1} aplica-se um processo iterativo. O novo ponto pode ser escolhido de alguma forma conveniente, como por exemplo, aleatoriamente dentro de um cubo multidimensional com lado de pequena dimensão δ , em torno do ponto X_n .
- Sorteando-se um possível ponto X_t , esta solução é aceita ou rejeitada considerando-se a

razão:

$$r = \frac{f_X(X_t)}{f_X(X_n)} \quad (4.31)$$

- Se $r > 1$, então o ponto X_t é aceito ($X_{n+1} = X_t$), enquanto que se $r < 1$, o ponto X_t é aceito com probabilidade r . Este procedimento é realizado comparando-se r com um número u uniformemente distribuído no intervalo $[0, 1]$, aceitando-se X_t , se $u < r$. Quando o ponto X_t não é aceito o caminho aleatório permanece no ponto X_n ($X_{n+1} = X_n$).
- Gera-se o ponto X_{n+2} usando-se o mesmo procedimento.

O valor de δ deve ser escolhido de forma que 1/3 a 1/2 das configurações geradas sejam aceitas, caso contrário o método torna-se pouco eficiente (Klar et al., 1996). Se houver uma grande quantidade de configurações rejeitadas significa que o valor de δ é muito grande; caso contrário se δ é muito pequeno, há muitas configurações aceitas mas a região explorada pelo método é pequena. A melhor escolha para o ponto inicial X_0 encontra-se onde a distribuição de probabilidades corresponda a um máximo (Kaviski, 2006).

4.3.3 As equações de evolução para funções randômicas

Nesta seção o fluxo de tráfego veicular é descrito através de quantidades macroscópicas introduzidas anteriormente. Considerando o trabalho (Delitala et al., 2007) a estrutura matemática para o caso espacialmente homogêneo (dados pelas equações (4.8) e (4.9)) é:

$$\frac{\partial f_i}{\partial t} = \sum_{h=1}^n \sum_{k=1}^n \eta_{hk}[\rho] A_{hk}^i[\rho] f_h(t) f_k(t) - f_i(t) \sum_{k=1}^n \eta_{ik}[\rho] f_k(t). \quad (4.32)$$

Essas interações são distribuídas sobre uma zona de visibilidade com comprimento característico $\varepsilon > 0$, ou seja, um veículo localizado em uma posição x é supostamente afetado por outros veículos que encontram-se dentro desta zona de visibilidade. Três categorias de veículos são considerados:

- (i) o veículo candidato com velocidade v_h .

(ii) o veículo campo com velocidade v_k .

(iii) o veículo teste com velocidade v_i .

Este modelo é caracterizado pelas quantidades:

- $\eta_{hk}[\rho]$ representado a razão de interação.
- A_{hk}^i definindo a densidade de probabilidade que o veículo candidato com velocidade v_h ajusta a sua velocidade para v_i depois de interagir com um veículo trafegando com velocidade v_k . A_{hk}^i depende da densidade local:

$$A_{hk}^i[\rho] \geq 0, \sum_{i=1}^n A_{hk}^i[\rho] = 1, \forall h, k = 1, \dots, n, \text{ para todo } \rho. \quad (4.33)$$

Os autores Delitala et al. (2007) sugerem que $A_{hk}^i[\rho]$ seja modelado por uma tabela de jogos com o parâmetro fenomenológico $\alpha \in [0, 1]$ que representa as condições da estrada, sendo $\alpha = 0$ a pior estrada. Esta tabela considera três casos:

- Interação com um veículo mais rápido ($v_h < v_k$), quando o veículo candidato encontra um veículo campo mais rápido: o veículo candidato ou mantém sua velocidade corrente ou possivelmente acelera, dependendo do espaço livre a frente.
- Interação com um veículo mais lento ($v_h > v_k$), quando o veículo candidato encontra um veículo campo mais lento e supõe-se que não acelerando a probabilidade de passar depende do tráfego local.
- Interação com um veículo com velocidade igual ($v_h = v_k$), quando o veículo candidato e o veículo campo estão viajando a mesma velocidade.

Os efeitos destas três interações são modelados por:

$$A_{hk}^i = \left\{ \begin{array}{l} 1 - \alpha(1 - \rho), \quad \left\{ \begin{array}{l} \text{se } v_i = v_h \text{ para } v_h < v_k, \\ \text{ou} \\ \text{se } v_i = v_k \text{ para } v_h > v_k, \\ \text{ou} \\ \text{se } v_i = v_h = 1 \end{array} \right. \\ \\ \alpha(1 - \rho) \quad \left\{ \begin{array}{l} \text{se } v_i = v_h + 1(v_h, v_k = 1, \dots, n) \text{ para } v_h < v_k, \\ \text{ou} \\ \text{se } v_i = v_h(v_h, v_k = 1, \dots, n) \text{ para } v_h > v_k, \\ \text{ou} \\ \text{se } v_i = 2 \text{ e } v_h = 1 \\ \text{ou} \\ \text{se } v_i = v_h + 1 \text{ para } v_h = 2, \dots, n - 1 \end{array} \right. \\ \\ \alpha\rho \quad \left\{ \begin{array}{l} \text{se } v_i = v_h - 1 \text{ para } v_h = 2, \dots, n - 1, \\ \text{ou} \\ \text{se } v_i = n - 1 \text{ para } v_h = n \end{array} \right. \\ \\ 1 - \alpha \quad \{ \text{se } v_i = v_h \text{ para } v_h = 2, \dots, n - 1 \\ \\ 1 - \alpha\rho \quad \{ \text{se } v_i = n \text{ para } v_h = n \\ \\ 0 \quad \text{caso contrário} \end{array} \right. \quad (4.34a)$$

Considerando a equação (4.32) e a densidade de probabilidade A_{hk}^i modelada pela tabela de jogos 4.34a (Delitala et al., 2007), a solução numérica é obtida pelo método de Runge-Kutta de 4^a. ordem.

4.3.4 Método de Runge-Kutta de 4ª Ordem

Os métodos de Runge-Kutta (Zill et al., 2001) são muito utilizados, principalmente porque podem ser expressos por uma sequência de fórmulas explícitas; sua implementação em computadores também é extremamente simples. O método de Runge-Kutta de quarta ordem consiste em determinar constantes apropriadas tais que uma fórmula como

$$y_{n+1} = y_n + ak_1 + bk_2 + ck_3 + dk_4 \quad (4.34)$$

concorde com um desenvolvimento de Taylor até h^4 , isto é, até o quinto termo, sendo k_1, k_2, k_3, k_4 múltiplos de $f(x, y)$ calculados em pontos selecionados. As fórmulas para a solução de $\frac{dy}{dx} = f(x, y)$ são as seguintes:

$$k_1(x, y) = f(x, y),$$

$$k_2(x, y) = f\left(x + \frac{h}{2}, y + \frac{h}{2}k_1(x, y)\right),$$

$$k_3(x, y) = f\left(x + \frac{h}{2}, y + \frac{h}{2}k_2(x, y)\right),$$

$$k_4(x, y) = f(x + h, y + hk_3(x, y)), \quad (4.35)$$

$$y_{n+1} = y_n + \frac{h}{6}[k_1(x, y) + 2k_2(x, y) + 2k_3(x, y) + k_4(x, y)].$$

Note que k_2 depende de k_1 , k_3 depende de k_2 e assim por diante. Também k_2 e k_3 envolvem aproximações do coeficiente angular no ponto médio do intervalo entre x_n e $x_{n+1} = x_n + h$. Outro atrativo dos métodos de Runge-Kutta é a fácil aplicação a sistemas de EDOs. Para um sistema de duas equações, $\frac{dx}{dt} = F(t, x, y)$ e $\frac{dy}{dt} = G(t, x, y)$, por exemplo:

$$k_1(t, x, y) = F(t, x, y),$$

$$m_1(t, x, y) = G(t, x, y),$$

$$k_2(t, x, y) = F\left(t + \frac{h}{2}, x + \frac{h}{2}k_1(t, x, y), y + \frac{h}{2}m_1(t, x, y)\right),$$

$$m_2(t, x, y) = G\left(t + \frac{h}{2}, x + \frac{h}{2}k_1(t, x, y), y + \frac{h}{2}m_1(t, x, y)\right),$$

$$k_3(t, x, y) = F\left(t + \frac{h}{2}, x + \frac{h}{2}k_2(t, x, y), y + \frac{h}{2}m_2(t, x, y)\right),$$

$$m_3(t, x, y) = G\left(t + \frac{h}{2}, x + \frac{h}{2}k_2(t, x, y), y + \frac{h}{2}m_2(t, x, y)\right),$$

$$k_4(t, x, y) = F(t + h, x + hk_3(t, x, y), y + hm_3(t, x, y)),$$

$$m_4(t, x, y) = G(t + h, x + hk_3(t, x, y), y + hm_3(t, x, y)), \quad (4.36)$$

$$x_{n+1} = x_n + \frac{h}{6}[k_1(t, x, y) + 2k_2(t, x, y) + 2k_3(t, x, y) + k_4(t, x, y)],$$

$$y_{n+1} = y_n + \frac{h}{6}[m_1(t, x, y) + 2m_2(t, x, y) + 2m_3(t, x, y) + m_4(t, x, y)].$$

O método de Runge-Kutta-Fehlberg, descrito nas expressões (4.36), permite variar o passo h a cada iteração, de modo a melhorar a aproximação (Zill et al., 2001).

4.3.5 Análise das distribuições de probabilidade na geração das condições iniciais

Dos 18 tipos de geradores abordados anteriormente foram escolhidos cinco geradores de números aleatórios (rknuth, rdesrg, ranmpb, rcomb e ranlux) utilizados para gerar a função de distribuição inicial f_i , com o objetivo de resolver a equação (4.32) para o caso espacialmente homogêneo através do método de Runge-Kutta de quarta ordem, obtendo a velocidade média v e o fluxo médio q usando a tabela de jogos para A_{hk}^i dada por (4.33). A densidade adimensional

ρ varia de $[0, 1]$ e o parâmetro fenomenológico α assume os valores $\alpha = 1$ (melhor pista), $\alpha = 0,6$ (pista de qualidade média) e $\alpha = 0,3$ (pista de qualidade baixa).

Os resultados estão descritos nas Figuras 4.22 e 4.23. Foram escolhidos os mesmos geradores para as variações na qualidade da pista. Comparando os resultados entre as figuras quanto ao parâmetro fenomenológico α , constata-se que quanto maior for a qualidade da estrada menor é a influência dos geradores na função de distribuição inicial, mesmo para o gerador *ranlux*, que é considerado o mais estável e confiável.

Para a Figura 4.22 utiliza-se três valores do parâmetro fenomenológico α e denota-se com subíndices os valores da velocidade v_a e v_b referindo-se a velocidade $v(DL)$ e $v(ranlux)$ para $\alpha = 1, 0$; v_c e v_d referindo-se a velocidade $v(DL)$ e $v(ranlux)$ para $\alpha = 0,6$ e v_e e v_f referindo-se a velocidade $v(DL)$ e $v(ranlux)$ para $\alpha = 0,3$.

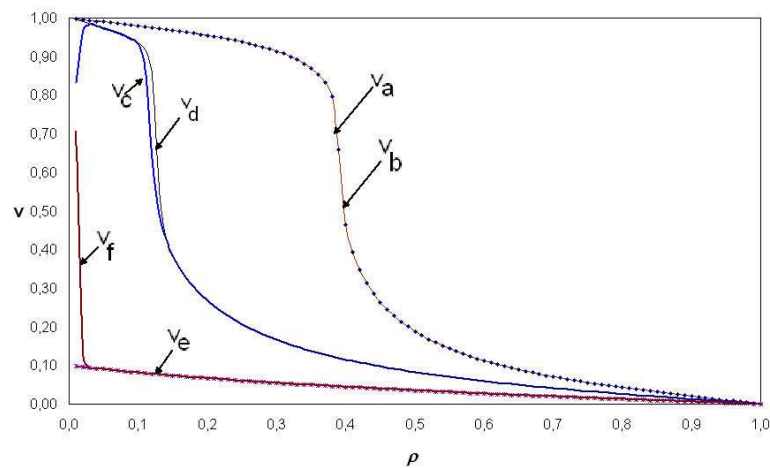


Figura 4.22: Velocidade média v versus a densidade ρ apêndice A, anexo (5).

Para a figura 4.23 utiliza-se três valores do parâmetro fenomenológico α e denota-se com subíndices os valores do fluxo q_a e q_b referindo-se ao fluxo $q(DL)$ e $q(ranlux)$ para $\alpha = 1, 0$; q_c e q_d referindo-se ao fluxo $q(DL)$ e $q(ranlux)$ para $\alpha = 0,6$ e q_e e q_f referindo-se ao fluxo $q(DL)$ e $q(ranlux)$ para $\alpha = 0,3$

Observa-se que para a melhor estrada ($\alpha = 1, 0$) e considerando-se uma precisão da ordem de 10^{-3} , quaisquer um dos geradores escolhidos podem ser utilizados.

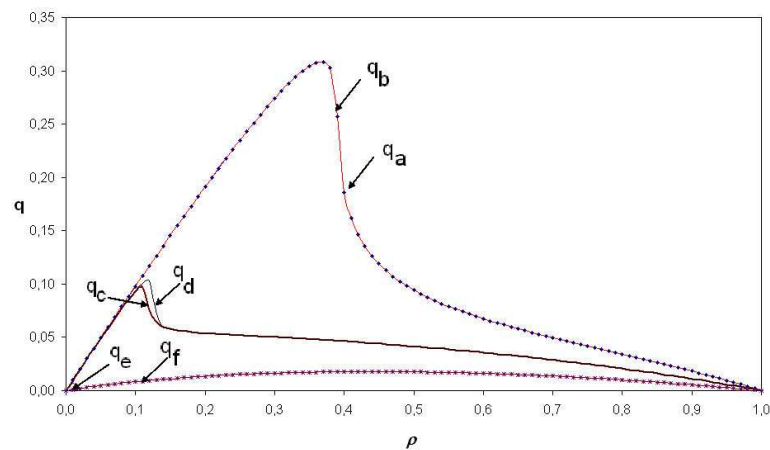


Figura 4.23: Fluxo médio q versus a densidade ρ , apêndice A, anexo (5).

4.3.6 Conclusões da análise das funções aleatórias

Neste seção apresenta-se uma análise dos efeitos das condições iniciais estabelecidas por diferentes métodos de geração de números pseudo-aleatórios uniformes, sobre a resolução da equação (4.32) para o caso espacialmente homogêneo obtendo a velocidade média v e o fluxo médio q usando a tabela de jogos modelada pela equação (4.33).

Para tanto foram investigados 18 algoritmos para geração de números pseudo-aleatórios uniformes embora nas figuras apresentam os resultados para apenas dois deles. Considerou-se os dois principais: o original do artigo do Delitala et al.(2007) e o ranlux que é um gerador de números pseudo-aleatórios de alta qualidade muito utilizado. Estes algoritmos são fundamentados nas seguintes técnicas: (i) gerador congruente multiplicativo; (ii) gerador por deslocamento de registro; (iii) gerador com defasagens de Fibonacci; (iv) aleatorização por embaralhamento; (v) geradores combinados; e (vi) gerador RANLUX. As propriedades estatísticas dos números pseudo-aleatórios devem coincidir com as propriedades estatísticas dos números verdadeiramente aleatórios descritos na seção 4.3.

Os resultados obtidos para a densidade adimensional ρ entre $[0, 1]$ apresentou poucas flutuações para uma qualidade ótima de estrada. Para a estrada de qualidade baixa, as diferenças entre os valores das velocidades e dos fluxos para a densidade mais baixa ($\rho = 0,01$) são praticamente sete vezes maiores do que para outras densidades. Para a estrada de qualidade média

as diferenças da velocidade ocorrem a partir da terceira casa, exceto quando $\rho = 0,01$, cuja a diferença ocorre na primeira casa e as diferenças do fluxo ocorrem a partir da terceira casa.

Conclui-se que ao se considerar uma precisão da ordem de 10^{-3} , qualquer um dos geradores escolhido pode ser utilizado.

5 *Conclusões e Recomendações*

Estudou-se os problemas relacionados a simulação do fluxo de tráfego veicular em duas escalas: macroscópica e cinética. Os resultados são inéditos e alguns foram publicados em congressos.

Na escala macroscópica simulou-se o fluxo em uma pista circular utilizando-se o método das características e o método das diferenças finitas explícito. Ambos os métodos mostraram-se adequados para solucionar o problema proposto.

No segundo problema de uma pista com uma entrada lateral utilizou-se o método das diferenças finitas explícito e notou-se que tanto a densidade veicular quanto a velocidade média sofrem alterações. O destaque fica para a densidade que se estabiliza com o passar do tempo.

Utilizou-se o modelo das velocidades discretas na escala cinética. Considerou-se um sistema veículo-motorista e analisou-se o comportamento do motorista como uma partícula ativa. Conclui-se que a qualidade da pista influencia fortemente o fluxo e a velocidade média. O comportamento do motorista altera-se dependendo das condições iniciais mas não houve alteração no fluxo nem na velocidade média.

Também na escala cinética e para o caso espacialmente homogêneo investigou-se dezoito algoritmos para a geração de números pseudo-randômicos. Os resultados obtidos para a densidade ρ entre $[0, 1]$ apresentou poucas flutuações para uma qualidade ótima de pista. Para a pista de qualidade média as diferenças da velocidade ocorrem a partir da terceira casa, exceto quando $\rho = 0,01$, que a diferença ocorre na primeira casa e as diferenças do fluxo ocorrem a partir da terceira casa.

E no caso das pistas de qualidade baixa, as diferenças entre os valores das velocidades e dos fluxos para a densidade mais baixa ($\rho = 0,01$) são praticamente sete vezes maiores do que para outras densidades. Conclui-se que ao se considerar uma precisão da ordem de 10^{-3} , qualquer um dos geradores escolhido pode ser utilizado.

Uma extensão desta linha de pesquisa para trabalhos futuros seria:

- Considerar o caso espacialmente não-homogêneo (quando há variação com a posição x);
- A questão da qualidade da pista sendo alterada ao longo da simulação;
- Considerar duas ou mais vias na pista.

6 *Nomenclaturas e normas das rodovias em geral*

Neste capítulo apresentam-se informações gerais sobre o Código de Trânsito Brasileiro e estradas de rodagem.

6.1 Código de Trânsito Brasileiro

Segundo a lei 9.503 de setembro de 1997, nos artigos: 2º, 60º e Anexo I tem-se:

Art. 2º. São vias terrestres urbanas e rurais as ruas, as avenidas, os logradouros, os caminhos, as passagens, as estradas e as rodovias, que terão seu uso regulamentado pelo órgão ou entidade com circunscrição sobre elas, de acordo com as peculiaridades locais e as circunstâncias especiais.

Parágrafo único. Para os efeitos deste Código, são consideradas vias terrestres as praias abertas à circulação pública e as vias internas pertencentes aos condomínios constituídos por unidades autônomas.

Art. 60º. as vias abertas à circulação, de acordo com sua utilização, classificam-se em:

I-vias urbanas: a) via de trânsito rápido;

b) via arterial;

c) via coletora;

d) via local;

II-vias rurais: a) rodovias;

b) estradas.

Anexo I- dos conceitos e definições VIA: superfície por onde transitam veículos, pessoas e animais, compreendendo a pista, a calçada, o acostamento, ilha e canteiro central.

VIA DE TRÂNSITO RÁPIDO: aquela caracterizada por acessos especiais com trânsito livre, sem interseções em nível, sem acessibilidade direta aos lotes lindeiros e sem travessia de pedestres em nível.

VIA ARTERIAL: aquela caracterizada por interseções em nível, geralmente controlada por semáforo, com acessibilidade aos lotes lindeiros e às vias secundárias e locais, possibilitando o trânsito entre as regiões da cidade.

VIA COLETORA: aquela destinada a coletar e distribuir o trânsito que tenha necessidade de entrar ou sair das vias de trânsito rápidas ou arteriais, possibilitando o trânsito dentro das regiões da cidade.

VIA LOCAL: aquela caracterizada por interseções em nível não semaforizadas, destinadas apenas ao acesso local ou a áreas restritas.

VIA RURAL: estradas e rodovias.

VIA URBANA: ruas, avenidas, vielas, ou caminhos e similares abertos à circulação pública, situados na área urbana, caracterizados principalmente por possuírem imóveis edificadas ao longo da sua extensão.

VIAS E ÁREA DE PEDESTRES: vias ou conjuntos de vias destinadas à circulação prioritária de pedestres.

6.2 Estradas de Rodagem

Do livro sobre estradas de rodagem, projeto geométrico de Glauco Pontes Filho, Instituto Panamericano de Carreteras Brasil.

(1) Quanto à posição geográfica:

As estradas federais no Brasil recebem o prefixo BR, acrescido de três algarismo. O

primeiro algarismo tem o seguinte significado:

- 0 → rodovias radiais
- 1 → rodovias longitudinais
- 2 → rodovias transversais
- 3 → rodovias diagonais
- 4 → rodovias de ligação

Os dois outros algarismos indicam a posição da rodovia com relação à capital federal e aos limites extremos do país, de acordo com o seguinte critério:

RADIAIS: partem de Brasília, ligando as capitais e principais cidades. Têm a numeração de 010 a 080, obedecendo o sentido horário. Ex: BR-040 (Brasília-Rio de Janeiro).

LONGITUDINAIS: têm direção geral norte-sul. A numeração varia da direita para a esquerda, entre 100 e 199. Em Brasília o número é 150. Ex: BR-116 (Fortaleza-Jaguarão).

TRANSVERSAIS: têm direção geral leste-oeste, sendo caracterizadas pelo algarismo 2. A numeração varia de 200 no extremo norte do país a 250 em Brasília, indo até 299 no extremo sul. Ex: BR-230 (Transamazônica).

DIAGONAIS PARES: têm direção geral noroeste-sudeste (NO-SE). A numeração varia de 300 no extremo nordeste do país a 398 no extremo sudoeste (350 em Brasília). O número é obtido de modo aproximado, por interpolação. Ex: BR-316 (Belém-Maceió).

DIAGONAIS IMPARES: têm direção geral nordeste-sudoeste (NE-SO), e a numeração varia de 301 no extremo noroeste do país a 399 no extremo sudeste. Em Brasília o número é 351. Ex: BR-319 (Manaus-Porto Velho).

LIGAÇÕES: em geral essas rodovias ligam pontos importantes das outras categorias. A numeração varia de 400 a 450 se a ligação estiver para o norte de Brasília e, 451 a 499, se

para o sul de Brasília. Embora sejam estradas de ligação, chegam a ter grandes extensões, como a BR-407, com 1251 km. Já a BR-488 é a menor de todas as rodovias federais, com apenas 1 km de extensão. Esta rodovia faz a conexão da BR-116 com o Santuário Nacional de Aparecida, no estado de São Paulo.

No estado de São Paulo, as estradas são classificadas apenas em longitudinais e transversais. São longitudinais as rodovias que interligam pontos do interior do estado à capital, ou que estão alinhadas em direção à capital, e são codificadas por SP e um número que é o azimuth do alinhamento médio, aproximado para número par. São transversais aquelas que apenas interligam pontos no interior, não alinhados com a direção da capital, e são codificadas por SP e um número correspondente à distância média da rodovia até a cidade de São Paulo, aproximada para valor ímpar.

(2) Quanto à função:

A classificação funcional rodoviária é o processo de agrupar rodovias em sistemas e classes, de acordo com o tipo de serviço que as mesmas proporcionam e as funções que exercem. Quanto à função, as rodovias classificam-se em:

ARTERIAIS: proporcionam alto nível de mobilidade para grandes volumes de tráfego. Sua principal função é atender ao tráfego de longa distância, seja internacional ou interestadual.

COLETORAS: atende a núcleos populacionais ou centros geradores de tráfego de menor vulto, não servidos pelo Sistema Arterial. A função deste sistema é proporcionar mobilidade e acesso dentro de uma área específica.

LOCAIS: constituído geralmente por rodovias de pequena extensão, destinadas basicamente a proporcionar acesso ao tráfego intra-municipal de áreas rurais e de pequenas localidades às rodovias mais importantes.

(3) Quanto à jurisdição:

FEDERAIS: é, em geral, uma via arterial e interessa diretamente à Nação, quase sempre percorrendo mais de um estado. São construídas e mantidas pelo governo federal.

ESTADUAIS: são as que ligam entre si cidades e a capital de um estado. Atende às necessidades de um estado, ficando contida em seu território. Têm usualmente a função arterial ou coletora.

MUNICIPAIS: são as construídas e mantidas pelo governo municipal. São do interesse de um município ou de municípios vizinhos, atendendo ao município que a administra, principalmente.

VICINAIS: são em geral estradas municipais, pavimentadas ou não, de uma só pista, locais, e de padrão técnico modesto. Promovem a integração demográfica e territorial da região na qual se situam e possibilitam a elevação do nível de renda do setor primário. Podem também ser privadas, no caso de pertencerem a particulares.

(4) Quanto às condições técnicas:

As principais características geralmente consideradas nesse tipo de classificação são aquelas que se relacionam diretamente com a operação do tráfego (velocidade, rampas, raios, larguras de pista e acostamento, distância de visibilidade, níveis de serviço, etc.). Estas, por sua vez, são restringidas por considerações de custo, condicionados especialmente pelo relevo.

O tráfego, cujo atendimento constitui a principal finalidade da rodovia, é um dos elementos fundamentais a considerar. Recomenda-se adotar, como critério para a classificação técnica de rodovias, o volume de tráfego que deverá utilizar a rodovia no 10^o ano após sua abertura ao tráfego.

Além do tráfego, a importância e a função da rodovia constituem elementos para seu enquadramento em determinada classe de projeto.

As classes de projeto recomendadas encontram-se resumidas na tabela abaixo a seguir.

| Classe de projeto | Características | Critério de classificação técnica |
|-------------------|--|--|
| 0 | Via expressa Controle total de acesso | Decisão administrativa |
| I(A) | Pista dupla Controle parcial de acesso | Os volumes de tráfego previstos ocasionarem níveis de serviço em rodovia de pista simples inferiores aos níveis C ou D |
| I(B) | Pista simples Controle parcial de acesso | Volume horário de projeto > 200 Volume médio diário (VDM) > 1400 |
| II | Pista simples | VDM entre 700 e 1400 |
| III | Pista simples | VDM entre 300 e 700 |
| IV(A) | Pista simples | VDM entre 50 e 200 |
| IV(B) | Pista simples | VDM ≤ 50 |

Fonte: DNER

VDM=Volume médio diário previsto no ano de abertura ao tráfego

Níveis de serviço

O conceito de nível de serviço está associado às diversas condições de operação de uma via, quando ela acomoda diferentes volumes de tráfego.

O nível de serviço é estabelecido em função da velocidade desenvolvida na via e da relação entre o volume de tráfego e a capacidade da via.

Qualquer seção de uma via pode operar em diferentes níveis de serviço, dependendo do instante considerado. De acordo com o **Highway Capacity Manual**, os diversos níveis de serviço são assim definidos:

NÍVEL A: condição de escoamento livre, acompanhada por baixos volumes e altas densidades.

A densidade do tráfego é baixa, com velocidade controlada pelo motorista dentro dos limites de velocidade e condições físicas da via. Não há restrições devido à presença de

outros veículos.

NÍVEL B: fluxo estável, com velocidades de operação a serem restringidas pelas condições de tráfego. Os motoristas possuem razoável liberdade de escolha da velocidade e ainda têm condições de ultrapassagem.

NÍVEL C: fluxo ainda estável, porém as velocidades e as ultrapassagens já são controladas pelo alto volume de tráfego. Portanto, muitos dos motoristas não têm liberdade de escolher faixa e velocidade.

NÍVEL D: próximo à zona de fluxo instável, com velocidades de operação toleráveis, mas consideravelmente afetadas pelas condições de operação, cujas flutuações no volume e as restrições temporárias podem causar quedas substanciais na velocidade de operação.

NÍVEL E: é denominado também de nível de capacidade. A via trabalha a plena carga e o fluxo é instável, sem condições de ultrapassagem.

NÍVEL F: descreve o escoamento forçado, com velocidades baixas e volumes abaixo da capacidade da via. Formam-se extensas filas que impossibilitam a manobra. Em situações extremas, velocidade e fluxo podem reduzir-se a zero.

Referências

- ABBOT, M. B. *Computational Hydraulics*. London: Pitman, 1980.
- ANDERSON, S. L. Random numbers generators on vector supercomputers and other advanced architectures. *SIAM Review*, v. 32, p. 221, 1990.
- BARONE, E. *Transp. Theory Stat. Phys.*, v. 9, p. 59, 1981.
- BELLOMO, N.; COSCIA, V. First order models and closure of the mass conservation equation in the mathematical theory of vehicular traffic flow. *CR Mecanique*, v. 333, p. 843–851, 2005.
- BELLOMO, N.; COSCIA, V.; DELITALA, M. On the mathematical theory of vehicular traffic flow i - fluid dynamic and kinetic modeling. *Math. Mod. Meth. Appl. Sci*, v. 12, p. 1801–1844, 2002.
- BIRD, G. *Molecular Gas Dynamic and the Direct Simulation of Gas Flows*. 1a. ed. Oxford: Claredon Press, 1994.
- CERN. 2002. Ranlux. Disponível em:
<<http://wwwasdoc.web.cern.ch/wwwasdoc/shortwrupsdir/v115/top.html>>.
- COMPAQ COMPUTER CORPORATION. *Compaq Fortran, Language Reference Manual*. [S.l.], 1999.
- COSCIA, V.; DELITALA, M.; FRASCA, P. On the mathematical theory of vehicular traffic flow models ii. discrete velocity kinetic models. *International Journal Non-linear Mechanics*, v. 42, p. 411–421, 2007.
- COURANT, R.; FRIEDRICHS, K.; LEWY, H. On the partial difference equations of mathematical physics. *IBM Journal*, p. 215–234, 1967.
- DAGANZO, C. Requiem for second order fluid approximations of traffic flow. *Transpn. Res. B*, v. 29B, p. 277–286, 1995.
- DAGANZO, C. A behavioral theory of multi-lane traffic flow i: Long homogeneous freeway. *Sections. Research Report UCB-ITS-RR-99-5*, University of California, Berkeley, CA, 1999.
- DARBHA, S.; RAJAGOPAL, K. Limit of a collection of dynamical systems: An application to modeling the flow of traffic. *Math. Mod. & Meth. in Appl. Sci.*, v. 12, p. 1381–1399, 2002.
- DELITALA, M.; TOSIN, A. Mathematical modeling of vehicular traffic: a discrete kinetic theory approach. *Math. Mod. & Meth. in Appl. Sci.*, no. 6, v. 17, p. 901–932, 2007.
- FUSHIMI, M.; TEZUKA, S. The k-distribution of generalized feedback shift register pseudorandom numbers. *Commun. ACM* 26, v. 7, p. 516, 1983.

- HARTEN, A. High resolution schemes for hyperbolic conservation laws. *J. Comput. Phys.*, v. 49, p. 357–393, 1983.
- HELBING, D. *Physica A*, v. 219, p. 391, 1995.
- HELBING, D. Gas-kinetic derivation of navier-stokes-like traffic equations. *Phys. Rev.*, v. 53, p. 1067–1141, 1996.
- HELBING, D. Traffic and related self-driven many-particle systems. *Review of Modern Physics*, v. 73, p. 1067–1741, 2001.
- HENDERSON, F. *Open Channel Flow*. [S.l.]: Macmillan, 1966.
- HUANG, K. *Statistical Mechanics*. 2a. ed. New York: J. Wiley & Sons, 1987.
- HUTCHINSON, D. A new uniform pseudorandom number generator. *Commun. ACM* 9, v. 6, p. 432, 2004.
- INTERNATIONAL BUSINESS MACHINES CORPORATION. *IBM System/360: Scientific Subroutine Package, Version III. Programmer's manual*. [S.l.], 1970.
- KAVISKI, E. Representação de números de ponto flutuante. In: *Anais do XXIV Congresso Nacional de Matemática Aplicada*. Belo Horizonte: [s.n.], 2001.
- KAVISKI, E. *Solução de problemas de fenômenos de transporte pelo método de Monte Carlo*. 382 p. Tese (Doutorado) — PPGMNE, Universidade Federal do Paraná, Curitiba, 2006.
- KERNER, B. *The Physics of Traffic*. New York: Springer, 2004.
- KLAR, A.; KÜHNE, R.; WEGENER, R. Mathematical models for vehicular traffic. *Surveys Math. Ind.*, v. 6, p. 215–239, 1996.
- KNUTH, D. *The art of computer programming, vol. 2, Seminumerical Algorithms*. [S.l.]: Addison-Wesley, 1981.
- KREMER, G. M. *Uma Introdução à Equação de Boltzmann*. [S.l.]: UFPR, 2003.
- LAHEY COMPUTER SYSTEM, INC. *Fortran 95: Language Reference*. [S.l.], 1998.
- LEVEQUE, R. *Numerical Methods for Conservation Laws*. ETH Zurich: Birkhäuser Verlag, 1990.
- LEVEQUE, R. *Finite Volume Methods for Hyperbolic Problems*. Cambridge: Cambridge University Press, 2002.
- LEWIS, P.; GOODMAN, A.; MILLER, J. A pseudo-random number generator for the system/360. *IBM Stat. J.*, v. 8, p. 136–146, 1969.
- LIGGET, J.; CUNGE, J. Numerical methods of solutions of the unsteady flow in open channels. *Water Resources Publications*, I, 1975.
- LIGHTHILL, M.; F.R.S.; WHITHAM, G. On kinematic waves. i. flood movement in long rivers. *Proc. R. Soc. London*, A229, p. 281–316, 1955.
- MAXWELL, J. On the dynamical theory of gases. *Phil. Trans.*, London, v. 47, p. 49–68, 1867.

- METROPOLIS, N. et al. Equations of state calculations by fast computing machines. *Journal of Chemical Physics* 21, v. 6, p. 1087–1092, 1953.
- M.LÜSCHER. A portable high-quality random number generator for lattice field theory simulations. *Comp. Phys. Comm.* 79, v. 2, p. 100, 1994.
- NELSON, P. A kinetic model of vehicular traffic and its associated bimodal equilibrium solution. *Transp. Theory Stat. Phys.*, v. 24, p. 383–409, 1995.
- NEWMAN, M.; BARKEMA, G. *Monte Carlo Method in Statistical Physics*. Oxford: Clarendon Press, 1999.
- PAVERI-FONTANA, S. L. On boltzmann like treatments for traffic flow. *Transp. Res.*, v. 9, p. 225–235, 1975.
- PHILLIP, D.; REVINDRAN, A.; SOLBERG, J. *Operations Research: Principles and Practice*. New York: J. Wiley & Sons, 1975.
- PHILLIPS, W. *Transp. Plann. Technol.*, v. 5, p. 131, 1981.
- PHILLIPS, W. In: IEEE, 1981. *Proceedings of the 1978 IEEE Conference on Decisions and Control*. [S.l.]: IEEE, 1981. p. 1032.
- PIPES, L. *J. Appl. Phys.*, v. 24, p. 274, 1953.
- PRADO, A. L.; KAVISKI, E.; CUMIN, L. M. G. Teoria de tráfego de fluido para um sistema macroscópico. In: ABCM, XI., 2006. in *Proceedings of the 11th Brazilian Congress of Thermal Sciences and Engineering*. Curitiba,PR, 2006.
- PRADO, A. L.; KAVISKI, E.; CUMIN, L. M. G. Simulação para o fluxo de tráfego utilizando equações simplificadas da dinâmica dos fluidos. In: *Anais do X Encontro de Modelagem Computacional*. Petrópolis,RJ: [s.n.], 2007.
- PRADO, A. L.; KAVISKI, E.; CUMIN, L. M. G. Representação do sistema veículo-motorista através da partícula ativa para o estudo de tráfego veicular na escala cinética. In: *Anais do 30º CILAMCE -Congresso Ibero-Latino-Americano de Métodos Computacionais em Engenharia*. Armação de Búzios, RJ: [s.n.], 2009.
- PRESS, W. et al. *Numerical Recipes in Pascal*. Cambridge: Cambridge University Press, 1992.
- PRIGOGINE, I.; HERMAN, R. *Kinetic Theory of Vehicular Traffic*. New York: American Elsevier, 1971.
- RICHARD, P. *Oper. Res.*, v. 4, p. 42, 1956.
- SEMENZATO, R. *Transp. Theory Stat. Phys.*, v. 9, p. 83, 1981.
- SEMENZATO, R. *Transp. Theory Stat. Phys.*, v. 9, p. 95, 1981.
- SHCHUR, L.; BUTERA, P. *The RANLUX generator: resonances in a random walk test*. [S.l.]: Helplat 9805017 v3, 1998.
- SIVALOGANATHAN, K. Free surface flow computations by characteristics. *Journal of the Hydraulics Division*, v. 4, p. 543–556, 1974.

SIVALOGANATHAN, K. Flood routing by characteristics methods. *Journal of the Hydraulics Division*, v. 7, p. 1076–1091, 1978.

SMITH, G. *Numerical solution of partial differential equations: finite difference methods*. Third edition. USA: Oxford University Press, 1985.

STEINSTRASSER, C. *Método difusivo de Lax aplicado na solução das equações de Saint Venant*. Curitiba: [s.n.], 2005.

STREETER, V. *Mecânica dos Fluidos*. [S.l.]: McGraw-Hill do Brasil, 1997.

TRIOLA, M. *Introdução à Estatística*. 9. ed. Rio de Janeiro: LTC, 2005.

VELASCO, R.; JR., W. M. Navier-stokes equations for traffic flow. *Phys. Rev.*, v. 4, p. 1–9, 2005.

WHITHAM, G. *Linear and Nonlinear Waves*. [S.l.]: Wiley-Interscience, 1974.

WICHMANN, B.; HILL, D. An efficient and portable pseudo-random number generator. *Appl. Statis.*, v. 31, p. 188, 1982.

YU, L. et al. *Métodos Numéricos em Recursos Hídricos*, v. 4, 1999.

ZILL, D.; CULLEN, M. *Equações Diferenciais, volumes 1 e 2*. 3. ed. [S.l.]: Ed. Makron Books, 2001.

ANEXO A - Informações em Anexos

Anexo 1:Através da Teoria Cinética dos Gases, desenvolvida por Ludwig Boltzmann no final do século XIX, foram dados os primeiros passos para determinar o comportamento macroscópico de um fluido a partir de um modelo microscópico. No modelo microscópico, o movimento das partículas é descrito através das Leis de Newton, o que possibilitou relacionar as propriedades microscópicas e macroscópicas através de uma função conhecida por Equação de Boltzmann.

Anexo 2:Espaço de fase é o espaço multidimensional constituído pelos pontos que correspondem aos estados de cada partícula.

Anexo 3:os fundamentos da teoria cinética moderna foram estabelecidos por Maxwell (Maxwell, 1867) que incorporou ideias estatísticas para a distribuição de velocidades para moléculas em equilíbrio.

Anexo 4:A equação cinética representa o movimento oriundo da equação da energia cinética. A energia cinética E_k de uma partícula de massa m é igual ao trabalho total realizado para acelerá-la a partir do repouso até sua velocidade v ; é também igual ao trabalho realizado para desacelerá-la até atingir seu repouso; sua formula é $E_k = \frac{mv^2}{2}$.

Anexo 5:Fluido compressível é um fluído que responde com uma redução do seu volume ao ser submetido à ação de uma força, uma tensão por exemplo, que o comprime.

Anexo 6:Continuum para a matemática o continuum se relaciona com os números reais, mais precisamente, a reta.

Anexo 7: Modelos da hidrodinâmica de primeira ordem são obtidos com a equação da conservação da massa, ou seja, a conservação do número de veículos na estrada.

Anexo 8: Fechamento significa obter uma ou mais relações que auxiliam a resolução da equação ou sistema de equações.

Anexo 9: Fenomenológica é um termo usado em ciência para descrever um conjunto de hipóteses que procuram descrever e/ou interpretar as observações experimentais de um ou mais fenômenos sendo consistentes com uma lei ou teoria de caráter fundamental, mas não são, necessariamente, dela derivadas de forma direta. Os modelos fenomenológicos frequentemente fazem uso de relação empírica, ou seja, relações matemáticas baseadas apenas em resultados experimentais e não em teorias.

Anexo 10: A Segunda lei de Newton do movimento ou o princípio fundamental da dinâmica que estabelece uma proporcionalidade entre causa (força) e efeito (aceleração) dada pela equação $F_r = m \cdot a$.

Anexo 11: Gás ideal é um modelo teórico, um gás imaginário cujas moléculas não têm volume nem forças de repulsão ou atração. O seu calor específico é constante, independente da temperatura. É um gás que obedece às equações $p \cdot V/T = k$ e $p \cdot V = n \cdot R \cdot T$, com exatidão matemática, esta última equação é conhecida como equação de Clapeyron (Huang, 1987).

Anexo 12: Na teoria da probabilidade e na estatística, a variância de uma variável aleatória X é uma medida da sua dispersão estatística, indicando quão longe em geral os seus valores se encontram do valor esperado. É calculada como $\theta(x) = \text{var}(x) = E(x - E(x))^2$ sendo $E(x)$ o valor esperado da variável aleatória X .

Anexo 13: A covariância (σ^2) é uma medida de dependência linear entre as duas variáveis aleatórias.

APÊNDICE A - Programas em Anexos

Anexo 1: Programa do método das características (Fortran).

```
cc-----
cc Programa:  trafcar4 (usando as duas condições: estacionário e variável)
cc Autor:    Eloy Kaviski
cc Versão:   01.02 - 30/08/2006
cc Objetivo: Solução das equações do fluxo de tráfego (W=constante)
cc           pelo método das características.
cc Unidades: Quilômetro (km) e hora (h).
cc-----
```

```
PARAMETER      (npm=20000)

REAL*8  x(2,npm),t(2,npm),v(2,npm),ro(2,npm), tlx(npm),vlx(npm)

REAL*8  rolx(npm),xi(npm),ti(npm),vi(npm),roi(npm)

INTEGER*4    np(2),nlx,nx,  nxg,ntg,nxtg(npm),i

COMMON      /cmais/x,t,v,ro,tlx,vlx,rolx,xi,ti,vi,roi,np,nlx

REAL*8      tal,roc,v0,alfa,lx,tt,dx,mdx

COMMON      /dados/tal,roc,v0,alfa,lx,tt,dx,mdx,nx

REAL*8      ais,aos,bis,bos,ibis_bos,ialfa,ital,iroc

COMMON      /param/ais,aos,bis,bos,ibis_bos,ialfa,ital,iroc

REAL*8      xg(npm),tg(npm),dxg,vmax,romax

COMMON      /grafic/xg,tg,dxg,vmax,romax,nxg,ntg,nxtg

REAL*8      aux,xt0,tx0

INTEGER*1    caso
```



```

cc-----
      WRITE(*,*) 'Inicializacoes'

      OPEN(UNIT=13,FILE='trafcar5cl.lst',STATUS='replace')

cc Dados de entrada
cc Caso 1 - Velasco e Marques, Jr (Phys. Rev. E 72, 2005)
      caso=1
cc caso=2
cc caso 2 o ro variavel
      tal  = 1.0d0/120.0d0
      roc  = 140.0d0
      v0   = 80.0d0
      alfa = 100.0d0
      lx   = 12.0d0
      tt   = 0.1d0
cc alterei alfa para 100 estava 20 (os testes foram 100,200,240,300,360)
      nx = 800
cc o arquivo primeiro foi alfa 100 e nx 200
cc Parâmetros para traçado do gráfico
      nxg  = 101
      ntg  = 101
      vmax = 200.0d0
      romax = 200.0d0
      vmax = dlog(vmax)
      romax = dlog(romax)
cc Inicializações
      dxg = lx/(nxg-1)
      do i = 1, nxg
          xg(i)  = (i-1)*dxg
          nextg(i) = 0
      end do
      aux = tt/(ntg-1)

```

```

do i = 1, ntg
    tg(i) = (i-1)*aux
end do

OPEN(UNIT=10,FILE='trfcar6v.$$$',STATUS='replace',ACCESS='direct',
-    RECL=24)

aux = 0.0d0

do i = 1, nxg*npn
    WRITE(10,rec=i) aux,aux,aux
end do

CLOSE(10)

OPEN(UNIT=10,FILE='trfcar6v.$$$',STATUS='old',ACCESS='direct',
-    RECL=24)

dx = lx/nx
mdx = 0.5d0*dx
ialfa = 1.0d0/alfa
ital = 1.0d0/tal
iroc = 1.0d0/roc
ais = alfa + 1.0d0
aux = dsqrt(ais)*ialfa
ais = ais*ialfa
bis = aux - ialfa
bos = -aux - ialfa
aos = 0.5d0*(ais - aux)
ais = 0.5d0*(ais + aux)
ibis_bos = 1.0d0/(bis - bos)

WRITE(*,*) 'Metodo das caracteristicas'

cc Primeira C+

np(2) = 1
x (2,1) = lx
t (2,1) = 0.0d0

call cix(caso,lx,v(2,1),ro(2,1))

```

```

nlx      = 1
tlx (1) = t (2,1)
vlx (1) = v (2,1)
rolx(1) = ro(2,1)
xi (1) = 0.0d0
ti (1) = t (2,1)
vi (1) = v (2,1)
roi(1) = ro(2,1)
call ip_cmais()
OPEN(UNIT=8,FILE='trafcar6v.txt',STATUS='replace')
call gravar()
cc Cálculo das C+ que emanam do eixo x
do i = nx, 1, -1
    xt0 = (i-1)*dx
    call cal_cmais(0,xt0)
    call ip_cmais()
    if (MOD(i,2).ne.0) call gravar()
end do
cc Cálculo das C+ que emanam do eixo t
tx0 = 0.0d0
i    = 1
do WHILE (tx0.Lt.tt)
    i = i + 1
    call cal_cmais(i,tx0)
    call ip_cmais()
    if (MOD(i,2).ne.0) call gravar()
end do
if (MOD(i,2).eq.0) call gravar()
CLOSE( 8)
CLOSE(10)
call arq_graf()

```

```

CLOSE(13)

WRITE(*,*) 'Fim do processamento'

stop

cc 1000  FORMAT(2i10,4e16.8) era este
1000  FORMAT(1i10,";",1i10,";",1e16.8,";",1e16.8,";",1e16.8,";",1e16.8)

end

cc-----

      subroutine cix(caso,xx,vx,rox)
cc Condições iniciais (t=0)

      INTEGER*1      caso
      REAL*8         xx,vx,rox,a,b
      REAL*8         tal,roc,v0,alfa,lx,tt,dx,mdx
      INTEGER*4      nx
      COMMON          /dados/tal,roc,v0,alfa,lx,tt,dx,mdx,nx

      if (caso.eq.1) then
        rox= 28.0d0
      else
        a=1.0d0/cosh(2.0d0*(xx-6.0d0))
        b=1.0d0/cosh(2.0d0*(xx-7.0d0))
        rox = 28.0d0 + 2.0d0*(a*a-b*b)
      end if

      vx  = 84.0d0 + 0.84d0*dsin(6.28318530718d0*xx/lx)
cc nos utilizamos em b o valor 7 pois xo=6km e deltaxo=1km

      return

      end subroutine

cc-----

      subroutine cal_cmais(tp,p)
cc Determinação da C+ que emanda do eixo x (tp=0 e x=p) ou que
cc emana do eixo t (tp>0 e t=tlx(tp))

      PARAMETER      (npm=20000)

      INTEGER*1      caso

```

```

INTEGER*4      tp, np(2),nlx,nx,i,j,k
REAL*8  p, x(2,npm),t(2,npm),v(2,npm),ro(2,npm), tlx(npm),vlx(npm)
REAL*8  tlx(npm),vlx(npm),rolx(npm),xi(npm),ti(npm),vi(npm),roi(npm)
COMMON      /cmais/x,t,v,ro,tlx,vlx,rolx,xi,ti,vi,roi,np,nlx
REAL*8      tal,roc,v0,alfa,lx,tt,dx,mdx
COMMON      /dados/tal,roc,v0,alfa,lx,tt,dx,mdx,nx
REAL*8      xx,vxx,roxx,aux

np(1) = np(2)
np(2) = 2
do i = 1, np(1)
  x (1,i) = x (2,i)
  t (1,i) = t (2,i)
  v (1,i) = v (2,i)
  ro(1,i) = ro(2,i)
end do

if (tp .eq. 0) then
cc C+ que emanam do eixo x
  x(2,1) = p
  t(2,1) = 0.0d0
  call cix(caso,p,v(2,1),ro(2,1))
  xx = p + mdx
  call cix(caso,xx,vxx,roxx)
  call calcnpto(x(2,1),t(2,1),v(2,1),ro(2,1), xx,0.0d0,vxx,
-               roxx,x(2,2),t(2,2),v(2,2),ro(2,2))
  else
cc C+ que emanam do eixo t
  x (2,1) = 0.0d0
  t (2,1) = tlx (tp)
  v (2,1) = vlx (tp)
  ro(2,1) = rolx(tp)
  x (2,2) = xi (tp)

```

```

        t (2,2) = ti (tp)
        v (2,2) = vi (tp)
        ro(2,2) = roi(tp)
        p = tlx(tp)
    end if
    i = 1
    do while ((x(2,np(2)).lt.lx).and.(i.le.np(1)))
        j      = np(2)
        np(2) = np(2) + 1
        k      = np(2)
        call calcnpto(x(2,j),t(2,j),v(2,j),ro(2,j), x(1,i),t(1,i),
-               v(1,i),ro(1,i),x(2,k),t(2,k),v(2,k),ro(2,k))
        i = i + 1
    end do
    nlx = nlx + 1
cc WRITE(*,*) tp,nlx,np(2),t(2,1),t(2,np(2))
    xi (nlx) = x (2,np(2)) - lx
    ti (nlx) = t (2,np(2))
    vi (nlx) = v (2,np(2))
    roi(nlx) = ro(2,np(2))
    do i = np(2)-1, 1, -1
        if ( lx .ge. x(2,i) ) then
            j      = i + 1
            aux = (lx - x(2,i))/(x(2,j) - x(2,i))
            tlx (nlx) = t (2,i) + aux*(t (2,j) - t (2,i))
            vlx (nlx) = v (2,i) + aux*(v (2,j) - v (2,i))
            rolx(nlx) = ro(2,i) + aux*(ro(2,j) - ro(2,i))
            return
        end if
    end do
    return

```

end subroutine

cc-----

REAL*8 function ftir(vx,rox)

cc Termo do lado direito das Invariantes de Riemman

REAL*8 vx,rox,a,b

REAL*8 tal,roc,v0,alfa,lx,tt,dx,mdx

INTEGER*4 nx

COMMON /dados/tal,roc,v0,alfa,lx,tt,dx,mdx,nx

REAL*8 ais,aos,bis,bos,ibis_bos,ialfa,ital,iroc

COMMON /param/ais,aos,bis,bos,ibis_bos,ialfa,ital,iroc

a = ital*(v0 - vx)/vx

b = ialfa*rox*vx*(1.0d0 - iroc*rox)

ftir = a - b

return

end

cc-----

subroutine gravar()

cc Gravação dos resultados

PARAMETER (npm=20000)

REAL*8 x(2,npm),t(2,npm),v(2,npm),ro(2,npm), tlx(npm),vlx(npm)

REAL*8 tlx(npm),vlx(npm),rolx(npm),xi(npm),ti(npm),vi(npm),roi(npm)

INTEGER*4 np(2),nlx,nx,i

COMMON /cmais/x,t,v,ro,tlx,vlx,rolx,xi,ti,vi,roi,np,nlx

REAL*8 tal,roc,v0,alfa,lx,tt,dx,mdx

COMMON /dados/tal,roc,v0,alfa,lx,tt,dx,mdx,nx

do i = 1, np(2)

if (x(2,i).lt.lx) then

if (MOD(i,2).ne.0) WRITE(8,1000) nlx,i,x(2,i),t(2,i),v(2,i),ro(2,i)

else

WRITE(8,1000) nlx,i,lx,tlx(nlx),vlx(nlx),rolx(nlx)

return

```

        end if
    end do
return
cc 1000  FORMAT(2i10,4e16.8) era este, alterei para o que está abaixo
1000  FORMAT(1i10,";",1i10,";",1e16.8,";",1e16.8,";",1e16.8,";",1e16.8)
    end subroutine

```

cc-----

```

    subroutine calcnpto(xl,tl,vl,rol,xr,tr,vr,ror,
-
                                xm,tm,vm,rom)
    REAL*8      xl,tl,vl,rol,xr,tr,vr,ror,xm,tm,vm,rom
    REAL*8      ais,aos,bis,bos,ibis_bos,ialfa,ital,iroc
    COMMON      /param/ais,aos,bis,bos,ibis_bos,ialfa,ital,iroc
    REAL*8      eps/1.0d-5/,difx,dift,difv,difro,ftir, lvl,lvr,blrol
    REAL*8      aux,bux,al,bl,ar,br,xm1,tm1,vm1,rom1,blror,fl,fr,fm
    INTEGER*2    nit/100/,it

    lvl  = dlog(vl)
    lvr  = dlog(vr)
    blrol = bis*dlog(rol)
    blror = bos*dlog(ror)
    aux   = xr - xl
    bux   = lvl - lvr + blrol - blror

    fl = ftir(vl,rol)
    fr = ftir(vr,ror)
    xm  = 0.5d0*(xl  + xr)
    tm  = 0.5d0*(tl  + tr)
    vm  = 0.5d0*(vl  + vr)
    rom = 0.5d0*(rol + ror)

    difx = xm
    dift = tm
    difv = vm
    difro = rom

```



```

it = 0

do WHILE (((difx.GT.eps*xm).or.(dift.GT.eps*tm ).or.
-      (difv.GT.eps*vm).or.(difro.GT.eps*rom)).and.(it.lT.nit))

    xml = xm
    tml = tm
    vml = vm
    roml = rom

    al = ais*(vl + vm)
    ar = aos*(vr + vm)
    tm = (aux + al*tl - ar*tr)/(al - ar)
    xm = xl + al*(tm - tl)

cc WRITE(*,*) tm,xm

    fm = ftir(vm,rom)
    bl = 0.5e0*(fl + fm)*(tm - tl)
    br = 0.5e0*(fr + fm)*(tm - tr)
    rom = (bl - br + bux)*ibis_bos
    vm = bl + lvl + blrol - bis*rom
    rom = dexp(rom)
    vm = dexp(vm)

cc WRITE(*,*) rom, vm

cc stop

    difx = dabs(xml - xm)
    dift = dabs(tml - tm)
    difv = dabs(vml - vm)
    difro = dabs(roml - rom)

cc WRITE(*,*) vm, rom

    it = it + 1

end do

if (xm.lt.0.0d0) WRITE(13,*) 'C1 ',xm,tm,vm,rom
if (tm.lt.0.0d0) WRITE(13,*) 'C2 ',xm,tm,vm,rom
if (vm.lt.0.0d0) WRITE(13,*) 'C3 ',xm,tm,vm,rom

```

```

        if (rom.lt.0.0d0) WRITE(13,*) 'C4 ',xm,tm,vm,rom
cc WRITE(*,*) it
        return
    end subroutine

cc-----

    subroutine ip_cmais()
cc Interpolação polinomial de Lagrange
cc Cálculo de t, v e ro, para tabela de valores de grafic
        PARAMETER      (npm=20000)
        REAL*8          x(2,npm),t(2,npm),v(2,npm),ro(2,npm), tlx(npm),vlx(npm)
        REAL*8          tlx(npm),vlx(npm),rolx(npm),xi(npm),ti(npm),vi(npm),roi(npm)
        INTEGER*4        np(2),nlx, nxg,ntg,nxtg(npm), i,j,h,k,r,i1,i2,k1,k2
        COMMON           /cmais/x,t,v,ro,tlx,vlx,rolx,xi,ti,vi,roi,np,nlx
        REAL*8           xg(npm),tg(npm),dxg,vmax,romax
        COMMON           /grafic/xg,tg,dxg,vmax,romax,nxg,ntg,nxtg
        INTEGER*1        c/2/
        REAL*8           e(3),p
        k1 = 1 + jidint(x(c,1)/dxg)
        k2 = 2 + jidint(x(c,np(c))/dxg)
        if (k1.gt.nxg) k1 = nxg
        if (k2.gt.nxg) k2 = nxg
        do WHILE ((xg(k1).LT.x(c,1)).AND.(k1.LT.nxg))
            k1 = k1 + 1
        end do
        do WHILE ((xg(k2).GT.x(c,np(c))).AND.(k2.GT.1))
            k2 = k2 - 1
        end do
        h = 1
        do k = k1, k2
            do WHILE ((xg(k).GT.x(c,h)).AND.(h.LT.np(c)))
                h = h + 1
            end do
        end do
    end subroutine

```

```

        end do
i1 = h - 1
i2 = h
if (i1.lt.1) i1 = 1
nxtg(k) = nxtg(k) + 1
do i = 1, 3
    e(i) = 0.0d0
end do
do i = i1, i2
    p = 1.0d0
    do j = i1, i2
        if (j.ne.i) p = p*(xg(k) - x(c,j))/(x(c,i) - x(c,j))
    end do
    e(1) = e(1) + p*t (c,i)
    e(2) = e(2) + p*dlog(v (c,i))
    e(3) = e(3) + p*dlog(ro(c,i))
end do
if (e(2).gt.vmax) then
    WRITE(13,*) 'I ',x(c,1),x(c,np(c)),t(c,1),t(c,np(c))
    WRITE(13,*) ' ',i1,i2,np(c)
    WRITE(13,*) ' ',xg(k),e
    e(2) = 0.5d0*(v(c,i1)+v(c,i2))
    WRITE(13,*) ' ',xg(k),e
else
    e(2) = dexp(e(2))
end if
if (e(3).gt.romax) then
    WRITE(13,*) 'I ',x(c,1),x(c,np(c)),t(c,1),t(c,np(c))
    WRITE(13,*) ' ',i1,i2,np(c)
    WRITE(13,*) ' ',xg(k),e
    e(3) = 0.5d0*(ro(c,i1)+ro(c,i2))

```

```

        WRITE(13,*) ' ', xg(k), e
    else
        e(3) = dexp(e(3))
    end if
    r = (k-1)*npm + nxtg(k)
    WRITE(10,rec=r) e
end do

cc WRITE(*,*) k-1,nxtg(k-1)
    return
end subroutine

cc-----

    subroutine arq_graf()
cc Interpolação polinomial de Lagrange
cc Cálculo de v e ro, para os pares xg e tg
cc Preparação dos arquivos com as tabelas

    PARAMETER      (npm=20000)

    REAL*8      x(2,npm),t(2,npm),v(2,npm),ro(2,npm), tlx(npm),vlx(npm)
    REAL*8      tlx(npm),vlx(npm),rolx(npm),xi(npm),ti(npm),vi(npm),roi(npm)
    INTEGER*4      np(2),nlx, nxg,ntg,nxtg(npm), i,j,h,k,r,s,i1,i2
    COMMON          /cmais/x,t,v,ro,tlx,vlx,rolx,xi,ti,vi,roi,np,nlx
    REAL*8          xg(npm),tg(npm),dxg,vmax,romax
    COMMON          /grafic/xg,tg,dxg,vmax,romax,nxg,ntg,nxtg
    INTEGER*1      c/1/
    REAL*8          e(3),p

    WRITE(*,*) 'Preparacao dos arquivos'

    OPEN(UNIT=10,FILE='trfcar6v.$$$',STATUS='old',ACCESS='direct',
-      RECL=24)

    OPEN(UNIT=11,FILE='trafcarv6v.txt',STATUS='replace')
    OPEN(UNIT=12,FILE='trafcarr6v.txt',STATUS='replace')

    do h = 1, nxg
        r = (h-1)*npm

```

```

do i = 1, nxtg(h)
    r = r + 1
    READ(10,rec=r) t(c,i),v(c,i),ro(c,i)
cc WRITE(13,1000) xg(h),t(c,i),v(c,i),ro(c,i)
end do

s = 1
do k = 1, ntg
    do WHILE ((tg(k).GT.t(c,s)).AND.(s.LT.nxtg(h)))
        s = s + 1
    end do
    i1 = s - 1
    i2 = s
    if (i1.lt.1) i1 = 1
    do i = 2, 3
        e(i) = 0.0d0
    end do
    do i = i1, i2
        p = 1.0d0
        do j = i1, i2
            if (j.ne.i) p = p*(tg(k) - t(c,j))/(t(c,i) - t(c,j))
        end do
        e(2) = e(2) + p*dlog(v (c,i))
        e(3) = e(3) + p*dlog(ro(c,i))
    end do
    if (e(2).gt.vmax) then
        WRITE(13,*) 'G ',xg(h),t(c,1),t(c,nxtg(h))
        WRITE(13,*) ' ',i1,i2,nxtg(h)
        WRITE(13,*) ' ',tg(k),e(2),e(3)
        e(2) = 0.5d0*(v(c,i1)+v(c,i2))
        WRITE(13,*) ' ',tg(k),e(2),e(3)
    else

```

```

        e(2) = dexp(e(2))
    end if
    if (e(3).gt.romax) then
        WRITE(13,*) 'G ',xg(h),t(c,1),t(c,nxtg(h))
        WRITE(13,*) ' ',i1,i2,nxtg(h)
        WRITE(13,*) ' ',tg(k),e(2),e(3)
        e(3) = 0.5d0*(ro(c,i1)+ro(c,i2))
        WRITE(13,*) ' ',tg(k),e(2),e(3)
    else
        e(3) = dexp(e(3))
    end if

    WRITE(11,1000) xg(h),tg(k),e(2)
    WRITE(12,1000) xg(h),tg(k),e(3)
end do
end do

CLOSE(10)
CLOSE(11)
CLOSE(12)

return
1000  FORMAT(1e16.8,";",1e16.8,";",1e16.8,";",1e16.8)
cc 1000  FORMAT(4e16.8) era este

end subroutine

```

Anexo 2: Programa do método das diferenças finitas explícito (Fortran).

```

cc-----
cc Programa: ...for - Tráfego de Veículos em uma rodovia com condições
cc periodicas      Diferencas finitas - Metodo de Lax Wendorf
cc Equacao hiperbolica - Passo 1 e Passo 2
cc Modificado N e Nttot por problemas de arredondamento
cc Versao: 1

```

```

cc Autores: Adriana Luiza do Prado / Liliana Madalena Gramani / Eloy Kaviski
cc-----

      PARAMETER      (npm=30000)

cc npm - número máximo de pontos de x e t, condição npm>N
cc-----

cc Descricao dos vetores

      REAL*8      F(2,-2:npm),s(2,-2:npm),u(3,2,-2:npm),t(1,npm)

      REAL*8      v(1,-2:npm),x(1,-2:npm)

      COMMON      /soluc/F,s,u,t,v,x

cc F - vetor representando  $F(1)=\rho \cdot V$  e  $F(2)=\rho \cdot V \cdot V + P$ 
cc s - vetor representando  $s(1)=0$  e  $s(2)=\rho \cdot (V_0 - V)/\tau - \rho \cdot (1-p) \cdot P$ 
cc u - vetor representando  $u(t,1)=\rho$  e  $u(t,2)=\rho \cdot V$ 
cc t - tempo de analise da rodovia
cc v - igual a vx, velocidade em uma determinada posicao na rodovia
cc      utilizado para gravacao
cc x - igual a xx, posicao na rodovia, utilizado para gravacao
cc Obs: Nas contas no caderno tem-se para i: i-1, i-1/2, i, i+1/2, i+1
cc      e no programa fica i-2, i-1, i, i+1, i+2, respectivamente.
cc-----

cc Dados iniciais

      REAL*8      alfa,roc,rox,v0,tal,lx,ttot,dx,dt,xx,vx

      INTEGER*4   N, Nttot,it,i,j

      COMMON      /dados/alfa,roc,rox,v0,tal,lx,ttot,dx,dt,it,N,Nttot,xx,vx

cc alfa - constante adimensional
cc roc - número de veículos por hora (veic/h)
cc rox - número de veículos por km para t=0 (veic/km)
cc v0 - velocidade inicial dos veículos (km/h)
cc tal - constante de relaxacao do tempo (h)
cc ttot - tempo total de analise (h)
cc lx - extensao da rodovia (km)
cc dx - incremento na posicao

```

```

cc dt      - incremento no tempo
cc N        - número total de pontos de x
cc Nttot    - número total de pontos para t
cc it       - contador utilizado para gravacao
cc i        - referente a x
cc j        - referente a componente do vetor
cc xx       - posicao do veículo na rodovia
cc vx       - velocidade do veículo na rodovia na posicao xx
cc Obs: o valor de alfa e determinado experimentalmente e em uma regioao
cc         de estabilidade da equacao (21) do artigo do Wilson
cc Obs: dx e dt devem ser pequenos para analise de estabilidade no
cc         sistema de equacoes para u, F, s, ou seja, o número de Coutant,
cc         pc, definido por  $pc = k/h = dt/dx < 1$  (condicao de estabilidade
cc Obs: A partir da escolha de N determina-se npm. Neste programa, usando o
cc         metodo de Lax Wendorf (dois passos) definiu-se  $N = lx/(dx/2)$  pois
cc         utiliza-se tambem as posicoes medias, isto e, "i", "i + dx/2" e
cc         "i + dx". Para lx e dx determinados como dados iniciais, obteve-se
cc          $N = 24000$ , sendo esta a razao para se escolher  $npm = 30000$ .
cc Obs: Nttot e definido como  $Nttot = ttot/dt$ , para as condicoes de contorno
cc  $Nttot = 10.000$ 
cc Obs: A gravacao ocorrera a cada 100 resultados, ou seja,
cc          $Nttot/"100" = 100$  gravacoes . "referente a mod(it.100)"
cc Obs: j = 1,2 representando as duas coordenadas dos vetores
cc-----
cc Parâmetros calculados
cc         REAL*8          ialfa,ital,iroc,p,p1,tp,r
cc         COMMON          /param/ialfa,ital,iroc,p,p1,tp,r
cc ialfa - inverso de alpha
cc ital  - inverso de tal
cc iroc  - inverso de roc
cc p     - parâmetro que depende de rho, calculado por  $p = 1 - \rho / roc$ 

```



```

cc p1      - parâmetro auxiliar de calculo,  $p1=(1-p)*ialfa$ 
cc tp      - parâmetro auxiliar de contador de tempo, inicial zero, final 1h
cc          igual a  $tp=it*dt$ 
cc -----
cc      Programa Principal
cc          OPEN(UNIT=8,FILE='ultimoarqvrho.txt',STATUS='replace')
cc      Dados iniciais (entradas, alfa é adimensional, roc é veiculos/hora,
cc      rox=rho(x,0) é veiculos/km, v0 é km/h, tal está em hora,
cc      ttot= tempo total de análise de 1 hora,  lx é 12 km.
cc          alfa=100
cc          roc=140
cc          rox=28
cc          v0=84
cc          tal=1.0d0/120
cc          ttot=1.0e0
cc          lx=12.0e0
cc          N=24000
cc          dx=(2*lx)/N
cc          r=0.01d0
cc          dt=r*dx
cc      dx e dt devem ser pequenos para analise de estabilidade,no. de Courant,
cc       $p= k/h=dt/dx < 1$  neste caso onde a equação é hiperbólica
cc      melhor fazer N inteiro para nao dar problema e o Nttot eh 10000
cc      N é 24.000 representando o número total de pontos de x e t por
cc      esta razão tivemos que definir  $npm=30000$ , pois o metodo de Lax Wendorf
cc      utiliza alem do i e i + dx, tambem o ponto médio entre eles.
cc      Nttot é 10.000 pontos para o tempo dos quais escolheremos alguns
cc      valores que serão gravados.
cc          Nttot=10000
cc          j=0
cc          ialfa = 1.0d0/alfa

```

```

    ital  = 1.0d0/tal
    iroc  = 1.0d0/roc
cc      Início do calculo usando loop em it
    WRITE(*,*) 'inicio do calculo'
    WRITE(8,*) 'calculo_do_trafego_veicular_usando_elementos_finitos'
    WRITE(*,*) 'Dados de entrada'
    WRITE(8,1000) alfa,roc,rox,tal
    WRITE(8,1500) lx,dx,dt
1000    FORMAT('Alfa=',e8.2,3x,'Rhol=',e8.2,3x,'Rho 0=',e8.2,3x,'Tau=',e10.4)
1500    FORMAT('Comprim.',e8.2,3x,'Delta x=',e8.2,3x,'Delta t=',e8.2)
    tp=0.0d0
    do it=1,Nttot
        call calc()
        tp=it*dt
        call gravar()
    end do
    WRITE(*,*) 'final'
    CLOSE(8)
    STOP
    END

cc -----
cc      inicio da subrotina ( calculo de u em t=0 )
cc -----

    subroutine calc()
    PARAMETER      (npm=30000)
    REAL*8          F(2,-2:npm),s(2,-2:npm),u(3,2,-2:npm),t(1,npm)
    REAL*8          v(1,-2:npm),x(1,-2:npm)
    COMMON          /soluc/F,s,u,t,v,x
    REAL*8          alfa,roc,rox,v0,tal,lx,ttot,dx,dt,xx,vx
    INTEGER*4       N, Nttot,it,i,j
    COMMON          /dados/alfa,roc,rox,v0,tal,lx,ttot,dx,dt,it,N,Nttot,xx,vx

```

```

REAL*8      ialfa,ital,iroc,p,p1,tp,r
COMMON      /param/ialfa,ital,iroc,p,p1,tp,r
cc      Condicoes iniciais (t=0)
cc      .....poderia-se usar i=-2,N+2 pois vx e periodico devido ao seno
IF(it.eq.1) then
cc      do i=-2,N+2,2
      do i=0,N
        xx=i*dx/2
        u(1,1,i)=rox
        vx=v0+0.84d0*dsin(6.28318530718d0*xx/lx)
        u(1,2,i)=rox*vx
        v(1,i)=vx
        x(1,i)=xx
      end do
      do i=1,2
        do j=1,2
          u(1,j,-i) = u(1,j,N-i)
          u(1,j,N+i)=u(1,j,i)
          v(1,-i)=u(1,2,-i)/u(1,1,-i)
        end do
      end do
      x(1,-2)=0
      x(1,-1)=dx/2
      call gravar()
      WRITE(*,*) 'testou e saiu do if com t=0'
END if
cc      ....condicoes iniciais para F e s
      do i=-2,N+2,2
        F(1,i)=u(1,2,i)
        vx=u(1,2,i)/u(1,1,i)
        F(2,i)= (1+ialfa)*(vx*u(1,2,i))

```

```

s(1,i)=0
p=1-(u(1,1,i)*iroc)
p1=(1-p)*ialfa
s(2,i)=u(1,1,i)*(v0-vx)*ital - u(1,2,i)*u(1,2,i)*p1
end do

cc      ...calculo de u(t=2,j,i+1)    (Passo 1)
do i=0,N,2
  do j=1,2
    a=0.5d0*(u(1,j,i+2)+u(1,j,i))
    b=0.5d0*r*(F(j,i+2)-F(j,i))
    c=(dt/4)*(s(j,i+2)+s(j,i-2))
    u(2,j,i+1)=a-b+c
  end do
end do

cc      ...condicoes periodicas (de contorno)
u(2,1,-1)=u(2,1,N-1)
u(2,2,-1)=u(2,2,N-1)
do i=-1,N+1,2
  do j=1,2
    end do
end do

cc      ...condições intermediárias para F e s
cc      do i=-2,N+2
cc      era para ser de dois em dois
do i=-1,N+1,2
  F(1,i)=u(2,2,i)
  vx=u(2,2,i)/u(2,1,i)
  F(2,i)= (1+ialfa)*(vx*u(2,2,i))
  s(1,i)=0
  p=1-(u(2,1,i)*iroc)
  p1=(1-p)*ialfa

```

```

        s(2,i)=u(2,1,i)*(v0-vx)*ital-u(2,2,i)*u(2,2,i)*p1
    end do
cc    ...cálculo de u(t=3,j,i)(Passo 2)
do i=0,N,2
    do j=1,2
        d= r*(F(j,i+1)-F(j,i-1))
        e= (dt/2)*(s(j,i+1)+s(j,i-1))
        u(3,j,i)= u(1,j,i)-d+e
    end do
end do
cc    ...condições periódicas (de contorno)
u(3,1,-2)=u(3,1,N-2)
u(3,2,-2)=u(3,2,N-2)
u(3,1,N+2)=u(3,1,2)
u(3,2,N+2)=u(3,2,2)
cc    colocamos so os pares
do i=-2,N+2,2
    v(1,i)=u(1,2,i)/u(1,1,i)
end do
cc    do i=-2,N+2
cc        do j=1,2
c        ...zerando os valores
cc            u(1,j,i)=0.d0
cc            u(2,j,i)=0.d0
cc            F(j,i)=0.d0
cc            s(j,i)=0.d0
cc            v(1,i)=0.d0
cc        end do
cc    end do
do i=-2,N+2,2
    do j=1,2

```

```

cc      ...para continuar o cálculo até lh assume-se os valores de
cc      u(3,...) para u(1,...) e volta a calcular
      u(1,j,i)=u(3,j,i)
      end do
      end do
      return
      END subroutine

cc -----
cc      subrotina gravar ( gravacao dos resultados)
cc -----

      subroutine gravar()
      PARAMETER      (npm=30000)
      REAL*8      F(2,-2:npm),s(2,-2:npm),u(3,2,-2:npm),t(1,npm)
      REAL*8      v(1,-2:npm),x(1,-2:npm)
      COMMON      /soluc/F,s,u,t,v,x
      REAL*8      alfa,roc,rox,v0,tal,lx,ttot,dx,dt,xx,vx
      INTEGER*4   N, Nttot,it,i,j
      COMMON      /dados/alfa,roc,rox,v0,tal,lx,ttot,dx,dt,it,N,Nttot,xx,vx
      REAL*8      ialfa,ital,iroc,p,p1,tp,r
      COMMON      /param/ialfa,ital,iroc,p,p1,tp,r
      IF(it.eq.1) then
        do i=0,N,120
          WRITE(8,4000) x(1,i),v(1,i),u(1,1,i)
          WRITE(*,*) 'chegou'
        END do
      END if
      IF(MOD(it,1000).ne.0) return
      WRITE(8,3000)
      do i=0,N,120
        WRITE(8,4000) tp,x(1,i),u(1,1,i),v(1,i)
      end do

```

```

        return
3000  FORMAT(3x,'t',4x,'x',20x,'rho',16x,'v')
4000  FORMAT(3x, 1e12.5,";",6x,1e12.5,";",3x,1e23.8,";",3x,1e16.8)
        end subroutine

```

Anexo 3: Programa para pista única com entrada lateral (Delphi).

Autor: Eloy Kaviski

```

unit trafil;

interface

uses

    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
    StdCtrls;

type

    TForm1 = class(TForm)
        Button1: TButton;
        Button2: TButton;
        Button3: TButton;
        procedure Button1Click(Sender: TObject);
        procedure Button2Click(Sender: TObject);
        procedure Button3Click(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    Form1: TForm1;

implementation

{$R *.DFM}

type

```

```

    vet0 = array[0..30000] of double;
    vet1 = array[1..2,0..30000] of double;
var
    ialfa,
    uialfa,
    iral,
    tal,
    iroc,
    xx,vx,
    dx,
    dts2,
    dt,t  : double;
    arq   : textfile;
var
    x,q : vet0;
    f,s : vet1;
    u   : array[1..2] of vet1;
const
    alfa : double = 100.0;
    roc  : double = 140.0;
    rox  : double = 28.0;
    v0   : double = 84.0;
    ital : double = 120.0;
    ttot : double = 1.0;
    lx   : double = 12.0;
    r    : double = 0.01;
    nx   : longint = 12000;
    nigx : word    = 200;
    nigt : word    = 2000;
procedure ler;
var

```



```

    arq    : textfile;
    s      : string;
    i,
    i1,i2  : byte;
    rc     : integer;
    s12    : array[1..12] of string;
begin
    assignfile(arq,'traf.dad');
    reset      (arq);
    i := 0;
    repeat
        readln(arq,s);
        i1 := pos('=',s);
        i2 := pos(';',s);
        i   := i + 1;
        s12[i] := copy(s,i1+1,i2-i1-1);
    until eof(arq);
    closefile(arq);
    val(s12[ 1],alfa,rc);
    val(s12[ 2],roc ,rc);
    val(s12[ 3],rox ,rc);
    val(s12[ 4],v0  ,rc);
    val(s12[ 5],ital,rc);
    val(s12[ 6],ttot,rc);
    val(s12[ 7],lx  ,rc);
    val(s12[ 8],r   ,rc);
    val(s12[ 9],nx  ,rc);
    val(s12[10],nigx,rc);
    val(s12[11],nigt,rc);
end;

procedure gravar;

```

```

var
    i : longint;
begin
    for i := 0 to nx do
        if (i mod nigx) = 0
        then
            begin
                vx := u[1][2,i]/u[1][1,i];
                writeln(arq,t:10:5,x[i]:10:4,u[1][1,i]:10:5,vx:10:4);
            end;
        end;
    procedure calc;
    var
        i : longint;
        a : byte;
        b : double;
    procedure calc_fs;
    begin
        vx := u[a][2,i]/u[a][1,i];
        f[1,i] := u[a][2,i];
        f[2,i] := u[1][2,i]*vx;
        s[2,i] := u[a][1,i]*(ital*(v0-vx)-u[a][2,i]*u[a][2,i]*iral)-vx*q[i];
    end;
    begin
        a := 1;
        for i := 0 to nx do { i*dx }
            calc_fs;
        for i := 1 to nx do { Predictor: (i-0.5)*dx }
            for a := 1 to 2 do
                u[2][a,i] := 0.5*(u[1][a,i-1] + u[1][a,i] -
                    r*(f[a,i] - f[a,i-1]) +

```

```

                dts2*(s[a,i-1] + s[a,i]));

a := 2;
for i := 1 to nx do
    calc_fs;
    for i := 1 to nx-1 do { Corrector: i*dx }
        for a := 1 to 2 do
            u[1][a,i] := u[1][a,i] - r*(f[a,i+1] - f[a,i]) +
                                dts2*(s[a,i] + s[a,i+1]);

b := pi*t;
u[1][1,0] := rox*(1.0 + 0.1*sin(b));
u[1][2,0] := v0*(1.0 - 0.1*sin(b))*u[1][1,0];
for a := 1 to 2 do
    u[1][a,nx] := 2.0*u[1][a,nx-1] - u[1][a,nx-2];
end;

procedure TForm1.Button1Click(Sender: TObject);

var
    aux    : double;
    i,
    nq,iq  : longint;
    j      : word;
begin
    ler;

    ialfa  := 1.0/alfa;
    uialfa := 1.0 + ialfa;
    iroc   := 1.0/roc;
    tal    := 1.0/ital;
    iral   := iroc*ialfa;
    dx     := lx/nx;
    dt     := r*dx;
    dts2   := 0.5*dt;
    aux    := 2.0*pi/lx;

```

```

nq := trunc(0.6*lx);
iq := trunc(8.0e-3/dx);
for i := 0 to nx do
    q[i] := 0.0;
for i := nq-iq to nq+iq do
    q[i] := 160.0;
for i := 0 to nx do
    begin
        xx := i*dx;
        vx := v0*(1.0 + 0.01*sin(aux*xx));
        x[i] := xx;
        s[1,i] := q[i];
        u[1][1,i] := rox;
        u[1][2,i] := rox*vx;
    end;
assignfile(arq,'traf1.txt');
rewrite    (arq);
t := 0.0;
gravar;
repeat
    for j := 1 to nigt do
        begin
            t := t + dt;
            calc;
        end;
    gravar;
{
    showmessage(floattostr(t));
}

until t > ttot;
closefile(arq);

```

```

        showmessage('Fim da execução.');
```

end;

```

function memolivre:longint;

var

    memo : tmemorystatus;

begin

    memo.dwLength := sizeof(memo);

    globalmemorystatus(memo);

    memolivre := memo.dwTotalPhys;

end;
```

```

procedure travar(memo:longint);

begin

    repeat

        until memolivre >= memo;

    end;
```

```

procedure TForm1.Button2Click(Sender: TObject);

var

    memo : longint;

begin

    memo := memolivre;

    winexec('blnotas.exe traf.dad',sw_show);

    travar(memo);

end;
```

```

procedure TForm1.Button3Click(Sender: TObject);

begin

    application.terminate;

end;
```

```

end.
```

Anexo 4:Programa para pista única com motorista (Fortran).

Autor: Liliana Madalena Gramani

cc Last change: ALP 15 Dec 2008 8:51 pm

program tveldisc7

cc Este programa cria os diagramas fundamentais

cc $u_j(\text{atividade})$ é igual a tres valores juntos, 0,5 , 0.75 e 1

cc com $\beta_{tj} = u_j(1-\rho)$ e $\eta = 1$

cc programa correto - TESTE DE P_j

cc para $g=0$, P_j é um numero inteiro:

cc $P_{j1}=1, P_{j2}=0, P_{j3}=0$, estrada toda com $u_j=0.5$

cc $P_{j1}=0, P_{j2}=1, P_{j3}=0$, estrada toda com $u_j=0.75$

cc $P_{j1}=0, P_{j2}=0, P_{j3}=1$, estrada toda com $u_j=1$

cc para $g=1$, P_j é um numero nao inteiro:

cc $P_{j1}=0.6, P_{j2}=0.3, P_{j3}=0.1$ estrada com $u_j=0.5(60\%), 0.75(30\%)$ e $1(10\%)$

cc $P_{j1}=0.1, P_{j2}=0.3, P_{j3}=0.6$ estrada com $u_j=0.5(10\%), 0.75(30\%)$ e $1(60\%)$

cc $P_{j1}=0.33, P_{j2}=0.33, P_{j3}=0.33$ estrada com $u_j=0.5(33\%), 0.75(33\%)$ e $1(33\%)$

cc para $g=2$ usamos distribuicao gauss (motoristas).

PARAMETER (nmax = 100, p = 3)

INTEGER n,i,k,h,m,mm,nfile,j,l,g/2/

INTEGER tsteps,rhosteps

INTEGER ifprint(9)

REAL f(1:nmax,1:nmax),kv(1:nmax,1:nmax),v(1:nmax)

REAL A(0:p,1:nmax,1:nmax,1:nmax),uj,eta(1:nmax,1:nmax)

REAL k1(1:nmax,1:nmax),k2(1:nmax,1:nmax),k3(1:nmax,1:nmax)

REAL k4(1:nmax,1:nmax),zeros(1:nmax,1:nmax)

REAL dt,drho,tfin,u,q,rho,Ev2,alpha,theta

CHARACTER*4 string

cc -----

cc MAIN PROGRAM

cc -----

write(*,*)

```

write(*,*) 'tveldisc7last.f - December, 2008'
write(*,*)
open(1,FILE='datin03.txt')
read(1,*) n,dt,tfin,rhosteps,alpha
read(1,'(a4)') string
read(1,*) (ifprint(i),i=1,9)
close(1)
do i = 1,nmax
cc    zeros(i) = 0.
do j = 1,p
    f(i,j) = 0.      !set all variables to zero
    k1(i,j) = 0.
    k2(i,j) = 0.
    k3(i,j) = 0.
    k4(i,j) = 0.
    kv(i,j) = 0.
    zeros(i,j) = 0.
end do
end do
do uj = 0.5,1.0,0.25
cc    uj = 0.5
do i = 1,nmax
do h = 1,nmax
do k = 1,nmax
    A(uj,i,h,k) = 0.
end do
end do
end do
end do
cc
do h = 1,nmax

```

```

do k = 1,nmax
    eta(h,k) = 0.
end do

end do

call velocity(n,v)      !generate the velocity grid
call collker(n,v,eta)   !generate the collision kernel
tsteps = int(tfin/dt)   !set the number of timesteps
drho = 1./rhosteps      !set the density step
nfile = 1000 !open the files where the results will be written
do i = 1,9
    if (ifprint(i).eq.1) then
        write(string,fmt='(i4)') i*nfile
        open(i,FILE='new'//string)
        write(i,*) ' $ DATA=CURVE2D'
        write(i,*) ' % xlabel = "rho"'
        write(i,*) ' % markertype = 13'
cc        write(i,*) ' % xmin = 0.'
cc        write(i,*) ' % xmax = 1.'
cc        if ((i.eq.1).or.(i.eq.2)) then
cc            write(i,*) ' % ymin = -0.1'
cc            write(i,*) ' % ymax = 1.1'
cc        endif
        write(i,*) ' % linetype = 0,uj= 0,5,0,75,1,alpha= 1.0,Bj = uj'
cc        write(i,*) ' % linelabel = ',i
cc        write(i,*) ' % grid = on'
        if (i.eq.1) then
            write(i,*) ' % ylabel = "v"'
            write(i,*) ' % toplabel = "Average velocity"'
        endif
        if (i.eq.2) then
            write(i,*) ' % ylabel = "q"'

```



```

        write(i,*) ' % toplabel = "Average flux"'
    endif

    if (i.eq.3) then
        write(i,*) ' % ylabel = "Theta"'
        write(i,*) ' % toplabel = "Variance of the velocity"'
    endif

    if (i.eq.4) then
        write(i,*) ' % ylabel = "v"'
        write(i,*) ' % toplabel = "Average velocity per rho const"'
    endif

    if (i.eq.5) then
        write(i,*) ' % ylabel = "v"'
        write(i,*) ' % toplabel = "Average velocity per rho const"'
    endif
    endif
end do

cc

if(g.eq.2) call p_gaussx()

cc

do mm = 0,rhosteps
    call ic(n,f,mm*drho)      !generate the initial condition
    write(*,*) 'rho = ',mm*drho
    rho= mm*drho

    do m = 1,tsteps      ! start to advance in time (Runge-Kutta 4)
        call tablegames(n,f,mm,drho,alpha,0.,zeros,A) !compute k1
        call tableau_rk(n,eta,A,f,0.,zeros,k1)
        call tablegames(n,f,mm,drho,alpha,dt/2.,k1,A) !compute k2
        call tableau_rk(n,eta,A,f,dt/2.,k1,k2)
        call tablegames(n,f,mm,drho,alpha,dt/2.,k2,A) !compute k3
        call tableau_rk(n,eta,A,f,dt/2.,k2,k3)
        call tablegames(n,f,mm,drho,alpha,dt,k3,A) !compute k4
    end do
end do

```

```

        call tableau_rk(n,eta,A,f,dt,k3,k4)
        do i = 1,n      ! compute the solution for the new timestep
            do j = 1,p
                f(i,j) = f(i,j)+dt/6.*(k1(i,j)+2.*k2(i,j)+2.*k3(i,j)+k4(i,j))
            end do
        end do

        rho= mm*drho
cc   if ((rho.eq.0.25)) then
cc       call plotuqt(n,f,v,m,rho,ifprint)
cc   endif
cc   if ((rho.eq.0.8)) then
cc       call plotuqt(n,f,v,m,rho,ifprint)
cc   endif

        end do

        call plotmtv(n,f,v,mm,drho,ifprint) !
cc compute the macroscopic quantities at the steady state and
cc write the results on files after close the files

        end do

        do i = 1,9
            if (ifprint(i).eq.1) then
                write(i,*) ' $ END'
                close(i)
            endif
        end do

        write(*,*) 'End of job'

        write(*,*)

        stop

        end

cc ----- end of the main program

cc -----

cc   Subroutines and functions

```

```

cc -----
      subroutine velocity(n,v)
cc  Generate the velocity grid
      PARAMETER (nmax = 100)
      REAL v(1:nmax)
      INTEGER n,i
      do i = 1,n
        v(i) = (i-1.)/(n-1.)
      end do
      return
      end

cc-----

      subroutine collker(n,v,eta)
cc  Generate the collision kernel
      PARAMETER (nmax = 100)
      REAL eta(1:nmax,1:nmax)
      REAL v(1:nmax)
      INTEGER h,k,n
      do h = 1,n
        do k = 1,n
cc      eta(h,k) = abs(v(h)-v(k))+1./(n-1.)
          eta(h,k) = 1.0
        end do
      end do
      return
      end

cc-----

      subroutine ic(n,f,rho0)
cc  Generate the initial condition on f
      PARAMETER (nmax = 100,p = 3)
      REAL f(1:nmax,1:nmax),prob(10)

```

```

      REAL rho0,rho1,rho2,rho3,s,soma
      REAL pj1,pj2,pj3
      INTEGER p1,p2,p3
      INTEGER i,n,seed,j,g
      INTEGER(4) time_array(3)
cc      seed=time8()

      g = 2

      p1 = 0

      p2 = 0

      p3 = 1
pj1 = 0.33
pj2 = 0.33
pj3 = 0.33
if (g.eq.0) then
      call itime(time_array)
      seed=time_array(1)+time_array(2)+time_array(3)
      s = 0.
if (p1.eq.1) then
      do i = 1,n
          j = 1
          if (i.eq.n) then
              f(i,j) = rho0-s
          else
              f(i,j) = ran1(seed)*(rho0-s)
              s = s+f(i,j)
          endif
      end do
      endif
if (p2.eq.1) then
do i = 1,n
      j = 2

```

```

    if (i.eq.n) then
      f(i,j) = rho0-s
    else
      f(i,j) = ran1(seed)*(rho0-s)
      s = s+f(i,j)
    endif
  end do
endif

if (p3.eq.1) then
do i = 1,n
  j = 3
  if (i.eq.n) then
    f(i,j) = rho0-s
  else
    f(i,j) = ran1(seed)*(rho0-s)
    s = s+f(i,j)
  endif
end do
endif

soma = 0.
do j = 1,p
  do i = 1,n
    soma = soma + f(i,j)
    write(*,*) 'f(i,j)=',f(i,j),'i=',i,'j=',j
  end do
end do

write(*,*) 'soma=',soma,'      ','rho0=',rho0
endif

cc      teste

IF(g.eq.2) then
  call p_gaussp(rho0,prob)

```

```

    pj1=prob(1)
    pj2=prob(2)
    pj3=prob(3)
    END if
cc  Distribution (g=1): rho1=pj1*rho  rho2=pj2*rho  and  rho3=pj3*rho
    if (g.ge.1) then
        rho1 = pj1*rho0
    call itime(time_array)
    seed=time_array(1)+time_array(2)+time_array(3)
    s = 0.
    do i = 1,n
        j = 1
        if ((i.eq.n).and.(j.eq.1)) then
            f(i,j) = rho1-s
        else
            f(i,j) = ran1(seed)*(rho1-s)
            s = s+f(i,j)
        endif
        write(*,*) 'f(1)', f(i,j),rho1
    end do
rho2 = pj2*rho0
    call itime(time_array)
    seed=time_array(1)+time_array(2)+time_array(3)
    s = 0.
    do i = 1,n
        j = 2
        if ((i.eq.n).and.(j.eq.2)) then
            f(i,j) = rho2-s
        else
            f(i,j) = ran1(seed)*(rho2-s)
            s = s+f(i,j)

```

```

        endif
        write(*,*) 'f(2)', f(i,j),rho2
    end do

    rho3 = pj3*rho0
    call itime(time_array)
seed=time_array(1)+time_array(2)+time_array(3)

    s = 0.
    do i = 1,n
        j = 3
        if ((i.eq.n).and.(j.eq.3)) then
            f(i,j) = rho3-s
        else
            f(i,j) = ran1(seed)*(rho3-s)
            s = s+f(i,j)
        endif
        write(*,*) 'f(3)', f(i,j),rho3
    end do

soma = 0.
do i = 1,n
    do j = 1,p
        soma = soma + f(i,j)
    end do
end do

write(*,*) 'soma=',soma,'      ','rho0=',rho0
    endif
    return
end

c -----
      subroutine tablegames(n,f,mm,drho,alpha,beta,w,A)
cc  Generate the table of games

      PARAMETER (nmax = 100,p = 3)

```

```

REAL f(1:nmax,1:nmax),kv(1:nmax,1:nmax),v(1:nmax)
REAL A(0:p,1:nmax,1:nmax,1:nmax),uj
REAL w(1:nmax,1:nmax)
REAL rho,drho,sumk,alpha,beta,eps
INTEGER mm,i,h,k,n,j
rho = mm*drho
if (beta.ne.0) then
    sumk = 0.
    do i = 1,n
        do j = 1,p
            sumk = sumk+w(i,j)
        end do
    end do
    rho = rho+beta*sumk
endif
do uj = 0.5,1.0,0.25
cc    uj = 0.5
    do h = 1,n
        do k = 1,n
            do i = 1,n
                if (h.lt.k) then
                    if (i.eq.h) then
                        A(uj,i,h,k) = 1.-alpha*uj*(1.-rho)*(1.-rho)
                    elseif (i.eq.h+1) then
                        A(uj,i,h,k) = alpha*uj*(1.-rho)*(1.-rho)
                    else
                        A(uj,i,h,k) = 0.
                    endif
                endif
            end do
        end do
    end do
    if (h.gt.k) then
        if (i.eq.h) then

```



```

        A(uj,i,h,k) = alpha*uj*(1.-rho)*(1.-rho)
elseif (i.eq.k) then
        A(uj,i,h,k) = 1.-alpha*uj*(1.-rho)*(1.-rho)
else
        A(uj,i,h,k) = 0.
endif
endif
if ((h.eq.k).and.(h.ne.1).and.(h.ne.n)) then
        if (i.eq.h-1) then
                A(uj,i,h,k) = alpha*uj*rho*(1.-rho)
        elseif (i.eq.h) then
                A(uj,i,h,k) = 1.-alpha*uj*(1.-rho)
        elseif (i.eq.h+1) then
                A(uj,i,h,k) = alpha*uj*(1.-rho)*(1.-rho)
        else
                A(uj,i,h,k) = 0.
        endif
endif
if ((h.eq.k).and.(h.eq.1)) then !impossible to brake
        if (i.eq.h+1) then
                A(uj,i,h,k) = alpha*uj*(1.-rho)*(1.-rho)
        elseif(i.eq.h) then
                A(uj,i,h,k) = 1.-alpha*uj*(1.-rho)*(1.-rho)
        else
                A(uj,i,h,k) = 0.
        endif
endif
if ((h.eq.k).and.(h.eq.n)) then !impossible to accelerate
        if (i.eq.h-1) then
                A(uj,i,h,k) = alpha*uj*rho*(1.-rho)
        elseif (i.eq.h) then

```

```

        A(uj,i,h,k) = 1.-alpha*uj*rho*(1.-rho)
    else
        A(uj,i,h,k) = 0.
    endif
endif
end do
end do
end do
return
end
cc-----
subroutine tableau_rk(n,eta,A,f,alpha,kv,w)
cc  Compute the Runge-Kutta tableau
PARAMETER (nmax = 100,p = 3)
REAL eta(1:nmax,1:nmax)
REAL A(0:p,1:nmax,1:nmax,1:nmax),uj
REAL f(1:nmax,1:nmax)
REAL kv(1:nmax,1:nmax)
REAL w(1:nmax,1:nmax)
REAL s,alpha
INTEGER n,i,h,k,j,l
do i = 1,n
    do j = 1,p
        if (j.eq.1) then                                !value of uj
            uj = 0.5
        endif
        if (j.eq.2) then
            uj = 0.75
        endif
        if (j.eq.p) then

```

```

        uj = 1.0
        endif
        s = 0.
        do h = 1,n
            do k = 1,n
                do l = 1,p
s = s+eta(h,k)*A(uj,i,h,k)*(f(h,j)+alpha*kv(h,j))*(f(k,l)+alpha*kv(k,l))
                end do
            end do
        end do
        w(i,j) = s
        s = 0.
        do k = 1,n
            do l = 1,p
                s = s+eta(i,k)*(f(k,l)+alpha*kv(k,l))
            end do
        end do
        s = f(i,j)*s
        w(i,j) = w(i,j)-s
    end do
end do
return
end

cc -----
      subroutine plotuqt(n,f,v,m,rho,ifprint)
cc      Output format for PLOTMTV
      PARAMETER (nmax = 100,p = 3)
      INTEGER n,i,m,ifprint(9),j
      REAL f(1:nmax,1:nmax),v(1:nmax)
      REAL rho,q,u,Ev2,theta
      q = 0.

```

```

u = 0.
Ev2 = 0.
theta = 0.
do i = 1,n
  do j = 1,p                !compute the flux and the expected
    q = q+v(i)*f(i,j)       !value of v 2
    Ev2 = Ev2+v(i)**2*f(i,j)
  end do
end do
if (rho.ne.0) then
  u = q/rho                  !compute the average velocity and its
  theta = Ev2/rho-u**2 !variance
endif
cc  if ((ifprint(4).eq.1).and.(rho.eq.0.25)) then !velocity
cc    write(4,*) m,u
cc  endif
  if ((ifprint(5).eq.1).and.(rho.eq.0.8)) then !velocity
    write(5,*) m,u
    write (*,*) 'passssssssssando'
  endif
if (ifprint(6).eq.1) then                !flux
  write(6,*) m,q
endif
  if ((ifprint(7).eq.1).and.(rho.ne.0)) then !variance
    write(7,*) m,theta
  endif
return
end
cc -----
  subroutine plotmtv(n,f,v,mm,drho,ifprint)
cc  Output format for PLOTMTV

```

```

PARAMETER (nmax = 100,p = 3)
INTEGER n,i,mm,ifprint(9),j
REAL f(1:nmax,1:nmax),v(1:nmax)
REAL rho,q,u,Ev2,theta,drho
rho = mm*drho
q = 0.
u = 0.
Ev2 = 0.
theta = 0.
do i = 1,n                                !compute the flux and the expected
  do j = 1,p
    q = q+v(i)*f(i,j)                    !value of v 2
    Ev2 = Ev2+v(i)**2*f(i,j)
  end do
end do
if (rho.ne.0) then
  u = q/rho                               !compute the average velocity and its
  theta = Ev2/rho-u**2                    !variance
endif
if ((ifprint(1).eq.1).and.(rho.ne.0)) then !velocity
  write(1,*) mm*drho,u
endif
if (ifprint(2).eq.1) then                 !flux
  write(2,*) mm*drho,q
endif
if ((ifprint(3).eq.1).and.(rho.ne.0)) then !variance
  write(3,*) mm*drho,theta
endif
return
end

```

cc-----

```

subroutine p_gaussx()
cc
    REAL*16      x(10)
    INTEGER      n
    COMMON       /blgauss/x,n
    REAL*16      igauss,p(10),sp
    INTEGER*2     i

    n      = 3
    p(1) = 0.50q0
    p(2) = 0.20q0
cc  p(3) = 0.15q0
    sp = 0.0q0
    do i = 1, n-1
        sp = sp + p(i)
        x(i) = igauss(sp)
    end do
    return
end

cc -----
subroutine p_gaussp(ro,p)
    REAL      ro,p(10)
    REAL*16    x(10)
    INTEGER    n
    COMMON     /blgauss/x,n
    REAL*16    gauss,z(10),sp,m,s,pj
    INTEGER*2   j

    m = ro
    s = qSQRT(1.0q0 - ro*ro)
    if (s .GT. 1.0q-7) then
        sp = 0.0q0
        do j = 1, n-1

```

```

        z(j) = (x(j) - m)/s
        pj    = gauss(z(j)) - sp
        p(j) = pj
        sp    = sp + pj
    end do

    p(n) = 1.0q0 - sp
    return
end if

x(n) = 1.0q500
i     = 1
do WHILE (m .gt. x(i))
    i = i + 1
end do

p(i) = 1.0
do j = 1, i-1
    p(j) = 0.0
end do

do j = i+1, n
    p(j) = 0.0
end do

return
end

```

cc-----

```

    REAL*16 function gauss(x)
cc    area sob a curva normal de -infinito ate x

    REAL*16      x
    REAL*16      b(5)/ 0.319381530,-0.356563782, 1.781477937,
-               -1.821255978, 1.330274429/,p/0.2316419/,
-               c/0.39894228040143267793/,t,y,z,q

    if (x .lt. 0.0q0) then
        y = -x

```

```

        else
            y = x
        end if
t = 1.0/(1.0 + p*y)
z = c*qEXP(-0.5*y*y)
q = z*t*(b(1) + t*(b(2) + t*(b(3) + t*(b(4) + t*b(5))))))
if (x .lt. 0.0q0) then
    gauss = q
    return
end if
gauss = 1.0q0 - q
return
end

```

cc -----

```

REAL*16 function igauss(p)
cc  Inversa da fda Normal - p=P(X<=x)
REAL*16      p,q,t,x
REAL*4      c(0:2)/2.515517,0.802853,0.010328/,
-           d(0:3)/1.000000,1.432788,0.189269,0.001308/
if (p .LT. 0.5q0) then
    q = p
else
    q = 1.0q0 - p
end if
t = -qlog(q)
t = qsqrt(t+t)
x = t - (c(0) + t*(c(1) + t*c(2)))/
-      (d(0) + t*(d(1) + t*(d(2) + t*d(3))))
if (p .LT. 0.5q0) then
    igauss = -x
else

```



```

        igauss = x
    end if
return
end

cc -----

    function ran1(idum)
cc    Generate (pseudo)random numbers
    INTEGER idum,IA,IM,IQ,IR,NTAB,NDIV
    REAL ran1,AM,EPS,RNMX
    PARAMETER (IA=16807,IM=2147483647,AM=1./IM,IQ=127773,IR=2836,
-           NTAB=32,NDIV=1+(IM-1)/NTAB,EPS=1.2e-7,RNMX=1.-EPS)
    INTEGER j,k,iv(NTAB),iy
    save iv,iy
    data iv /NTAB*0/, iy /0/
    if (idum.le.0.or.iy.eq.0) then
        idum=max(-idum,1)
        do 11 j=NTAB+8,1,-1
            k=idum/IQ
            idum=IA*(idum-k*IQ)-IR*k
            if (idum.lt.0) idum=idum+IM
            if (j.le.NTAB) iv(j)=idum
11        continue
            iy=iv(1)
        endif
        k=idum/IQ
        idum=IA*(idum-k*IQ)-IR*k
        if (idum.lt.0) idum=idum+IM
        j=1+iy/NDIV
        iy=iv(j)
        iv(j)=idum
        ran1=min(AM*iy,RNMX)

```

```

        return
    end

cc -----
cc  dados do arquivo datin03.txt
cc  6      1.E-2      3e3      100      0.4
cc  n      dt      Tfin      rhosteps alpha
cc  1 1 0 0 0 0 0 0 0
cc  fprint

```

Anexo 5: Programa para comparação da geração de funções randômicas (Fortran).

Autores: Adriana Luiza do Prado / Liliana Madalena Gramani / Eloy Kaviski

```

cc Last change:  ALP      8 Feb 2010      8:18 pm

        program veldisc0m

cc Este programa gera os diagramas fundamentais
cc alterado com outras funções randômicas

        PARAMETER (nmax = 100)

        INTEGER n,i,k,h,m,mm,nfile

        INTEGER tsteps,rhosteps

        INTEGER ifprint(9)

        REAL f(1:nmax),v(1:nmax),A(1:nmax,1:nmax,1:nmax)

        REAL eta(1:nmax,1:nmax)

        REAL k1(1:nmax),k2(1:nmax),k3(1:nmax),k4(1:nmax),zeros(1:nmax)

        REAL dt,drho,tfin,u,q,rho,Ev2,alpha,theta

        CHARACTER*4 string

cc -----

cc      MAIN PROGRAM

        write(*,*)

```

```

write(*,*) 'veldisc0n.f - February 08, 2010'
write(*,*)
open(1,FILE='datin01.txt')
read(1,*) n,dt,tfin,rhosteps,alpha
read(1,'(a4)') string
read(1,*) (ifprint(i),i=1,9)
close(1)

do i = 1,nmax                                !set all variables to zero
    f(i) = 0.
    k1(i) = 0.
    k2(i) = 0.
    k3(i) = 0.
    k4(i) = 0.
    zeros(i) = 0.
end do

do i = 1,nmax
    do h = 1,nmax
        do k = 1,nmax
            A(i,h,k) = 0.
        end do
    end do
end do

do h = 1,nmax
    do k = 1,nmax
        eta(h,k) = 0.
    end do
end do

call velocity(n,v)                            !generate the velocity grid
call collker(n,v,eta)                        !generate the collision kernel
tsteps = int(tfin/dt)                        !set the number of timesteps
drho = 1./rhosteps                          !set the density step

```

```

nfile = 1000          !open the files where the results will be written
do i = 1,9
  if (ifprint(i).eq.1) then
    write(string,fmt='(i4)') i*nfile
    open(i,FILE='del'//string)
    write(i,*) ' $ DATA=CURVE2D'
    write(i,*) ' % xlabel = "rho"'
    write(i,*) ' % markertype = 13'
    write(i,*) ' % linetype = 0'
    if (i.eq.1) then
      write(i,*) ' % ylabel = "v"'
      write(i,*) ' % toplabel = "Average velocity"'
    endif
    if (i.eq.2) then
      write(i,*) ' % ylabel = "q"'
      write(i,*) ' % toplabel = "Average flux"'
    endif
    if (i.eq.3) then
      write(i,*) ' % ylabel = "Theta"'
      write(i,*) ' % toplabel = "Variance of the velocity"'
    endif
  endif
end do

do mm = 0,rhosteps
  call ic(n,f,mm*drho) !generate the initial condition
  write(*,*) 'rho = ',mm*drho
  do m = 1,tsteps      !start to advance in time (Runge-Kutta 4)
    call tablegames(n,f,mm,drho,alpha,0.,zeros,A)!compute k1
    call tableau_rk(n,eta,A,f,0.,zeros,k1)
    call tablegames(n,f,mm,drho,alpha,dt/2.,k1,A)!compute k2
    call tableau_rk(n,eta,A,f,dt/2.,k1,k2)
  end do
end do

```

```

      call tablegames(n,f,mm,drho,alpha,dt/2.,k2,A)!compute k3
      call tableau_rk(n,eta,A,f,dt/2.,k2,k3)
      call tablegames(n,f,mm,drho,alpha,dt,k3,A) !compute k4
      call tableau_rk(n,eta,A,f,dt,k3,k4)
      do i = 1,n          ! compute the solution for the new timestep
        f(i) = f(i)+dt/6.*(k1(i)+2.*k2(i)+2.*k3(i)+k4(i))
      end do
    end do

    call plotmtv(n,f,v,mm,drho,ifprint)
cc compute the macroscopic quantities at the steady state
cc and write the results on file
    end do
    do i = 1,9      !close the files
      if (ifprint(i).eq.1) then
        write(i,*) ' $ END'
        close(i)
      endif
    end do
    write(*,*) 'End of job'
    write(*,*)
    stop
  end

cc ----- end of the main program
cc      subroutines and functions
      subroutine velocity(n,v)
cc      Generate the velocity grid
      PARAMETER (nmax = 100)
      REAL v(1:nmax)
      INTEGER n,i
      do i = 1,n
        v(i) = (i-1.)/(n-1.)

```

```

        end do
    return
end

cc -----

    subroutine collker(n,v,eta)
cc    Generate the collision kernel
    PARAMETER (nmax = 100)
    REAL eta(1:nmax,1:nmax),v(1:nmax)
    INTEGER h,k,n
    do h = 1,n
        do k = 1,n
c            eta(h,k) = abs(v(h)-v(k))+1./(n-1.)
            eta(h,k) = 1.0
        end do
    end do
    return
end

cc -----

    subroutine ic(n,f,rho0)
cc    Generate the initial condition on f
    PARAMETER (nmax = 100)
    REAL f(1:nmax),rho0,s
    INTEGER i,n,seed
    INTEGER(4) time_array(3)
c    seed=time8()
    call itime(time_array)
    seed=time_array(1)+time_array(2)+time_array(3)
    s = 0.
    do i = 1,n-1
cc        era esta
cc        f(i) = ran1(seed)*(rho0-s)

```

```

        f(i)=rkmath(seed)*(rho0-s)

        s = s+f(i)

    end do

    f(n) = rho0-s

    return

end

cc -----

        subroutine tablegames(n,f,mm,drho,alpha,beta,w,A)
cc  Generate the table of games

        PARAMETER (nmax = 100)

        REAL f(1:nmax),v(1:nmax),A(1:nmax,1:nmax,1:nmax)

        REAL w(1:nmax)

        REAL rho,drho,sumk,alpha,beta,eps

        INTEGER mm,i,h,k,n

        rho = mm*drho

        if (beta.ne.0) then

            sumk = 0.

            do i = 1,n

                sumk = sumk+w(i)

            end do

            rho = rho+beta*sumk

        endif

        do h = 1,n

            do k = 1,n

                do i = 1,n

                    if (h.lt.k) then

                        if (i.eq.h) then

                            A(i,h,k) = 1.-alpha*(1.-rho)

                        elseif (i.eq.h+1) then

                            A(i,h,k) = alpha*(1.-rho)

                        else

```

```

        A(i,h,k) = 0.
    endif
endif
if (h.gt.k) then
    if (i.eq.h) then
        A(i,h,k) = alpha*(1.-rho)
    elseif (i.eq.k) then
        A(i,h,k) = 1.-alpha*(1.-rho)
    else
        A(i,h,k) = 0.
    endif
endif
if ((h.eq.k).and.(h.ne.1).and.(h.ne.n)) then
    if (i.eq.h-1) then
        A(i,h,k) = alpha*rho
    elseif (i.eq.h) then
        A(i,h,k) = 1.-alpha
    elseif (i.eq.h+1) then
        A(i,h,k) = alpha*(1.-rho)
    else
        A(i,h,k) = 0.
    endif
endif
if ((h.eq.k).and.(h.eq.1)) then ! impossible to brake
    if (i.eq.h+1) then
        A(i,h,k) = alpha*(1.-rho)
    elseif(i.eq.h) then
        A(i,h,k) = 1.-alpha*(1.-rho)
    else
        A(i,h,k) = 0.
    endif
endif

```



```

endif
if ((h.eq.k).and.(h.eq.n)) then ! impossible to accelerate
  if (i.eq.h-1) then
    A(i,h,k) = alpha*rho
  elseif (i.eq.h) then
    A(i,h,k) = 1.-alpha*rho
  else
    A(i,h,k) = 0.
  endif
endif
endif
end do
end do
end do
return
end

cc -----
      subroutine tableau_rk(n,eta,A,f,alpha,v,w)
cc  Compute the Runge-Kutta tableau
      PARAMETER (nmax = 100)
      REAL f(1:nmax),v(1:nmax),A(1:nmax,1:nmax,1:nmax)
      REAL eta(1:nmax,1:nmax),w(1:nmax)
      REAL s,alpha
      INTEGER n,i,h,k
      do i = 1,n
        s = 0.
        do h = 1,n
          do k = 1,n
            s = s+eta(h,k)*A(i,h,k)*(f(h)+alpha*v(h))*(f(k)+alpha*v(k))
          end do
        end do
        w(i) = s
      end do

```

```

    s = 0.
    do h = 1,n
        s = s+eta(i,h)*(f(h)+alpha*v(h))
    end do
    s = f(i)*s
    w(i) = w(i)-s
end do
return
end

cc -----
      subroutine plotmtv(n,f,v,mm,drho,ifprint)
cc      Output format for PLOTMTV
      PARAMETER (nmax = 100)
      INTEGER n,i,mm,ifprint(9)
      REAL f(1:nmax),v(1:nmax)
      REAL rho,q,u,Ev2,theta,drho
      rho = mm*drho
      q = 0.
      u = 0.
      Ev2 = 0.
      theta = 0.
      do i = 1,n
          q = q+v(i)*f(i)
          Ev2 = Ev2+v(i)**2*f(i)
      end do
      if (rho.ne.0) then
          u = q/rho
          theta = Ev2/rho-u**2
      endif
      if ((ifprint(1).eq.1).and.(rho.ne.0)) then !velocity
          write(1,2000) mm*drho,u

```

```

endif
if (ifprint(2).eq.1) then                                !flux
  write(2,2000) mm*drho,q
endif
if ((ifprint(3).eq.1).and.(rho.ne.0)) then !variance
  write(3,2000) mm*drho,theta
endif
cc      alterei para imprimir ; nos dados (usei 1 e 2 )
2000  FORMAT(6x,1e12.5,";",4x,1e12.5,";",3x,1e12.5)
      return
      end

cc -----
      function ran1(idum)
cc      Generate (pseudo)random numbers (c1)
      INTEGER idum,IA,IM,IQ,IR,NTAB,NDIV
      REAL ran1,AM,EPS,RNMX
      PARAMETER (IA=16807,IM=2147483647,AM=1./IM,IQ=127773,IR=2836,
-          NTAB=32,NDIV=1+(IM-1)/NTAB,EPS=1.2e-7,RNMX=1.-EPS)
      INTEGER j,k,iv(NTAB),iy
      save iv,iy
      data iv /NTAB*0/, iy /0/
      if (idum.le.0.or.iy.eq.0) then
        idum=max(-idum,1)
        do 11 j=NTAB+8,1,-1
          k=idum/IQ
          idum=IA*(idum-k*IQ)-IR*k
          if (idum.lt.0) idum=idum+IM
          if (j.le.NTAB) iv(j)=idum
11      continue
          iy=iv(1)
        endif

```

```

      k=idum/IQ
      idum=IA*(idum-k*IQ)-IR*k
      if (idum.lt.0) idum=idum+IM
      j=1+iy/NDIV
      iy=iv(j)
      iv(j)=idum
      ranl=min(AM*iy,RNMX)
      return
    end

cc terminava aqui
cc os programas abaixo sao do Eloy
cc
cc Funcao lahey
cc-----
cc Geracao de números pseudo-aleatorios
cc MC[13967501,0,2**31-1;524287] - Gerador do Lahey/Fujitsu Fortran 95
cc REAL*4      lahey
cc do i = 1, 20
cc   WRITE(*,*) lahey()
cc end do
cc stop
cc end
cc
cc Funcao lahey
cc
cc
cc-----
cc Geracao de números pseudo-aleatorios
cc MC[13967501,0,2**31-1;524287] - Gerador do Lahey/Fujitsu Fortran 95
cc
cc      REAL*4 function lahey()

```

```

REAL*4      x
INTEGER*4    s(1)/524287/
LOGICAL*4    t/.true./
if( t ) then
    t = .false.
    call random_seed(put=s(1:1))
end if
call random_number(x)
lahey = x
return
end

cc
cc Funcao rkmath
cc-----
cc  Geracao de números pseudo-aleatorios
cc  MC[5**15,7261067085,2**35;2**31-1] (Knuth, 1969)
cc  Versao: Eloy Kaviski                (c2)
cc
cc  REAL*4 function rkmath()
c
INTEGER*8    c,k/452807053/,x/0/
INTEGER*4    b
INTEGER*2    y(5)/255,255,255,127,0/,w(5)/77,23,203,176,1/,
-            d(4)/0,0,0,16928/
INTEGER*1    a(8),h(2,5),g(8),i
REAL*8       m
cc      m = 2**35, w = 7261067085
equivalence  (c,a(1)),(b,a(2)),(y(1),h(1,1)),(m,d(1)),(x,g(1))
b = 0
do i = 1, 5
    c      = b + w(i) + k*y(i)

```

```

        h(1,i) = a(1)
        g(i)    = a(1)
        END do
        g(5)    = IAND(g(5),7)
        h(1,5) = g(5)
        rknuh = x/m
        return
    end

cc Funcao rcdcl1
cc-----
cc  Geracao de números pseudo-aleatorios - CDC-6500/6600
cc  MC[31623,0,2**48;3203431780337] (Phillips, Ravindran e
cc  Solberg, 1976)
cc  Versao: Eloy Kaviski
cc  Versao: I*4 - Divisao alternativa (Kaviski, 1997) (c3)
cc
    REAL*4 function rcdcl1()
    INTEGER*4      c,k
    INTEGER*2      y(6),d(4),f(4)
    INTEGER*1      a(4),h(2,6),g(8),i
    REAL*8         x,z
cc      z = 2**4
    equivalence    (c,a(1)),(y(1),h(1,1)),(z,d(1)),(x,g(1)),(x,f(1))
    data           k/31623/,y/241,207,105,219,233,2/,d/0,0,0,16432/
    c = 0
    do i = 1, 6
        c      = c + k*y(i)
        h(1,i) = a(1)
        g(i)    = a(1)
        a(1)    = a(2)
        a(2)    = a(3)

```

```

        a(3)    = 0
        END do
        f(4)    = d(4)
        rcdcl1 = x - z
        return
    end

cc Funcao rcdcl2

cc-----

cc  Geracao de números pseudo-aleatorios - CDC-6500/6600
cc  MC[31623,0,2**48;3203431780337] (Phillips, Ravindran e
cc  Solberg, 1976)
cc  Versao: Eloy Kaviski
cc  Versao: I*8 - Divisao alternativa (Kaviski, 1997)

        REAL*4 function rcdcl2()

        INTEGER*8      y/3203431780337/
        INTEGER*8      n
        INTEGER*4      k/31623/
        INTEGER*2      a(4),b(4),d(4)/0,0,0,16432/
        REAL*8         x,z
        equivalence    (y,a(1)),(z,d(1)),(n,x),(x,b(1))

cc          Geracao

        y          = k*y
        a(4)       = 0

cc          Divisao por 2**48

        n          = y
        b(4)       = d(4)
        rcdcl2 = x - z
        return
    end

cc Funcao rcdcl3

cc-----

```

```

cc  Geracao de números pseudo-aleatorios - CDC-6500/6600
cc  MC[31623,0,2**48;3203431780337] (Phillips, Ravindran e
cc  Solberg, 1976)
cc  Versao: Eloy Kaviski
cc  Versao: I*8 - Divisao convencional      (c4)
cc

```

```

      REAL*4 function rcdcl3()

      INTEGER*8      y/3203431780337/

      REAL*8          m

      INTEGER*4        k/31623/

      INTEGER*2        a(4),d(4)/0,0,0,17136/

cc      m = 2**48

      equivalence      (y,a(1)),(m,d(1))

cc      Geracao

      y                = k*y

      a(4)              = 0

cc      Divisao por 2**48

      rcdcl3 = y/m

      return

      end

```

```

cc Funcao rcdcl4

```

```

cc-----

```

```

cc  Geracao de números pseudo-aleatorios - CDC-6500/6600
cc  MC[31623,0,2**48;3203431780337] (Phillips, Ravindran e
cc  Solberg, 1976)
cc  Versao: Eloy Kaviski
cc  Versao: I*4 - Divisao alternativa (Kaviski, 1997)

```

```

      REAL*4 function rcdcl4()

      INTEGER*4        c,k/31623/

      INTEGER*2        d(4)/0,0,0,16432/,f(4)/0,0,0,16432/,b,e

      INTEGER*1        a(4),h(2,6),g(8),i

```



```

      REAL*8      x,z
      INTEGER*2    y(6)/241,207,105,219,233,002/
cc      z = 2**4 = 16
      equivalence (c,a(1)),(b,a(2)),(y(1),h(1,1)),(z,d(1)),(x,g(1)),
-              (x,f(1))
      b = 0
      do i = 1, 6
          c      = b + k*y(i)
          h(1,i) = a(1)
          g(i)    = a(1)
      END do
      rcdcl4 = x - z
      return
      end

cc Funcao rsieml
cc-----
cc  Geracao de números pseudo-aleatorios - Siemens 7760
cc  MC[11133510565745309,0,2**56;3203431780337] (Ahrens e Dieter,1988)
cc  Versao: Eloy Kaviski      (c5)
      REAL*4 function rsieml()
cc      y = 3203431780337
cc      m = 2**56
      INTEGER*2    y(7)/241,207,105,219,233,002,000/
      INTEGER*8    k/11133510565745309/,c,d,e
      INTEGER*1    i,a(9),h(2,7),g(8)
      REAL*8      m
      INTEGER*2    b(4)/0,0,0,17264/
      INTEGER*8    x/0/
      equivalence (c,a(1)),(d,a(2)),(m,b(1)),(x,g(1)),(y(1),h(1,1))
cc      Geracao
      d      = 0

```

```

do i = 1, 7
    c      = d + k*y(i)
    h(1,i) = a(1)
    g(i)   = a(1)
end do

cc      Divisao por 2**56

rsieml = x/m

return

end

cc Funcao rcrayl

cc-----

cc Geracao de números pseudo-aleatorios - CRAY-X-MP
cc MC[44485709377909,0,2**48;3203431780337] (Anderson, 1990)
cc Versao: Eloy Kaviski
cc Versao: Divisao alternativa (Kaviski, 1997) (c6)

REAL*4 function rcrayl()

INTEGER*8      c,b,e,k/z"2875a2e7b175"/

INTEGER*2      d(4)/0,0,0,16432/,f(4)/0,0,0,16432/

INTEGER*1      a(9),h(2,6),g(6),i

REAL*8         x,z

INTEGER*2      y(6)/241,207,105,219,233,002/

cc      z = 2**4 = 16

equivalence (c,a(1)),(b,a(2)),(y(1),h(1,1)),
-          (z,d(1)),(x,g(1)),(x,f(1))

b      = 0

do i = 1, 6

    c      = b + k*y(i)

    h(1,i) = a(1)

    g(i)   = a(1)

END do

rcrayl = x - z

```

```

        return
    end

cc Funcao rcdcx1
cc-----
cc  Geracao de números pseudo-aleatorios - CDC-6500/6600
cc  MC[31623,2178281829,2**48;3203431780337] (Phillips, Ravindran e
cc  Solberg, 1976)
cc  Versao: Eloy Kaviski
cc  Versao: I*4 - Divisao alternativa (Kaviski, 1997) (c7)
    REAL*4 function rcdcx1()
        INTEGER*4      c,k
        INTEGER*2      y(6),w(6),d(4),f(4)
        INTEGER*1      a(4),h(2,6),g(8),i
        REAL*8         x,z
cc      z = 2**4, w = 2178281829
        equivalence (c,a(1)),(y(1),h(1,1)),(z,d(1)),(x,g(1)),(x,f(1))
        data          k/31623/,y/241,207,105,219,233,2/,d/0,0,0,16432/,
-          w/101,241,213,129,0,0/
        c = 0
        do i = 1, 6
            c      = c + w(i) + k*y(i)
            h(1,i) = a(1)
            g(i)   = a(1)
            a(1)   = a(2)
            a(2)   = a(3)
            a(3)   = 0
        END do
        f(4)      = d(4)
        rcdcx1 = x - z
        return
    end

```

cc Funcao rcdcx2

cc-----

cc Geracao de números pseudo-aleatorios - CDC-6500/6600

cc MC[31623,2178281829,2**48;3203431780337] (Phillips, Ravindran e
cc Solberg, 1976)

cc Versao: Eloy Kaviski

cc Versao: I*8 - Divisao alternativa (Kaviski, 1997)

REAL*4 function rcdcx2()

INTEGER*8 y/3203431780337/

INTEGER*8 w/2178281829/

INTEGER*8 n

INTEGER*4 k/31623/

INTEGER*2 a(4),b(4),d(4)/0,0,0,16432/

REAL*8 x,z

equivalence (y,a(1)),(z,d(1)),(n,x),(x,b(1))

cc Geracao

y = k*y + w

a(4) = 0

cc Divisao por 2**48

n = y

b(4) = d(4)

rcdcx2 = x - z

return

end

cc Funcao rcdcx3

cc-----

cc Geracao de números pseudo-aleatorios - CDC-6500/6600

cc MC[31623,2178281829,2**48;3203431780337] (Phillips, Ravindran e
cc Solberg, 1976)

cc Versao: Eloy Kaviski

cc Versao: I*8 - Divisao convencional (c8)

```

REAL*4 function rcdcx3()
INTEGER*8      y/3203431780337/
INTEGER*8      w/2178281829/
REAL*8         m
INTEGER*4      k/31623/
INTEGER*2      a(4),d(4)/0,0,0,17136/
cc      m = 2**48
      equivalence  (y,a(1)),(m,d(1))
cc      Geracao
      y          = k*y + w
      a(4)       = 0
cc      Divisao por 2**48
      rcdcx3 = y/m
      return
      end
cc Funcao rlgml
cc-----
cc  Geracao de números pseudo-aleatorios - IBM-370
cc  MC[16807,0,2**31-1;524287] (Lewis, Goodman e Miller, 1969)
cc  Versao: Park e Miller (1988)
      REAL*4 function rlgml()
      INTEGER*4      k/16807/,m/2147483647/,
-                  q/127773/,r/2836/,y/524287/,l
      INTEGER*2      h
      REAL*4         mr/2147483647.0/
      h = y/q
      l = MOD(y,q)
      y = k*l - r*h
      if (y .le. 0) y = y + m
      rlgml = y/mr
      return

```

```

end

cc Funcao rlgm2
cc-----
cc Geracao de números pseudo-aleatorios - IBM-370
cc MC[16807,0,2**31-1;524287] (Lewis, Goodman e Miller, 1969)
cc Versao: IMSL (1978) - Nome original: ggubfs (c9)

REAL*4 function rlgm2()

REAL*8      mr/2147483647.0/

INTEGER*8    m/2147483647/,k/16807/

INTEGER*4    y/524287/

y = mod(k*y,m)

rlgm2 = y/mr

return

end

cc Funcao randu1
cc-----
cc Geracao de números pseudo-aleatorios - IBM-370
cc MC[65539,0,2**31;524287] (IBM, 1972)
cc Versao: IBM (1972) - Nome original: randu (c10)

REAL*4 function randu1()

INTEGER*4    y/524287/,m/2147483647/

REAL*8      mr/2147483647.0/

y = 65539*y

if (y .lt. 0) y = (y + m) + 1

randu1 = y/mr

return

end

cc Funcao randu2
cc-----
cc Geracao de números pseudo-aleatorios - IBM-370
cc MC[65539,0,2**31;524287] (IBM, 1972)

```

cc Versao: Eloy Kaviski

```

REAL*4 function randu2()

INTEGER*8      k/65539/,z

INTEGER*4      y/524287/

INTEGER*1      a(4)

REAL*8         mr/2147483648.0/

equivalence    (z,y),(y,a(1))

z      = k*y

a(4) = IAND(a(4),127)

randu2 = y/mr

return

end

```

cc Funcao rdesrg

cc-----

cc Geracao de números pseudo-aleatorios

cc SR[15,17,32;524287] (Anderson, 1990)

cc Versao: Eloy Kaviski

cc Versao: Divisao alternativa (kaviski, 1997) (c11)

```

REAL*4 function rdesrg()

INTEGER*4      a(2),t,y/524287/,d(2)/0,1093664768/

INTEGER*1      i/15/,j/-17/

REAL*8         x,z

cc      z = 2**20

equivalence    (z,d(1)),(x,a(1))

t = IEOR(y,ISHFT(y,i))

y = IEOR(t,ISHFT(t,j))

a(1) = y

a(2) = d(2)

rdesrg = x - z

end

```

```
cc Funcao ranmpb
```

```
cc-----
```

```
cc Geracao de números pseudo-aleatorios
```

```
cc LF[55,31,1e9,-;ma(1:55)] (Knuth, 1969)
```

```
cc Versao: Press e outros (1992) - Nome: ran3
```

```
cc Adotado para uso em DSMC, por Bird (1994), com o nome: rf (c12)
```

```

      REAL*4 function ranmpb()

      INTEGER*4      mbig/1000000000/,mseed/161803398/,mz/0/,
-                  mj,mk

      REAL*4          fac/1.0e-9/

      INTEGER*1        iff/0/,i,ii,k

      INTEGER*4        ma(55),inext,inextp

      if (iff .eq. 0) then

        iff          = 1

        mj           = mod(mseed,mbig)

        ma(55) = mj

        mk           = 1

        do 50 i = 1, 54

          ii          = mod(21*i,55)

          ma(ii)= mk

          mk          = mj - mk

          if (mk .lt. mz) mk = mk + mbig

          mj = ma(ii)

50      continue

        do 100 k = 1, 4

          do 60 i = 1, 55

            ma(i) = ma(i) - ma(1+mod(i+30,55))

            if (ma(i) .lt. mz) ma(i) = ma(i) + mbig

60      continue

100     continue

        inext        = 0

```



```

        inextp = 31
        end if
200  inext = inext + 1
        if (inext .eq. 56) inext = 1
        inextp = inextp + 1
        if (inextp .eq. 56) inextp = 1
        mj = ma(inext) - ma(inextp)
        if (mj .lt. mz) mj = mj + mbig
        ma(inext) = mj
        ranmpb      = mj*fac
        if (ranmpb .gt. 1.e-8 .and. ranmpb .lt. 0.99999999) return
        go to 200
    end

cc Funcao rcomb
cc-----
cc  Geracao de números pseudo-aleatorios
cc  {MC[1364,0,1500419;1]+MC[1528,0,1400159;3]} mod 1 (Sharipov, 2002)
cc  Versao: Eloy Kaviski                      rcomb(c13)

    REAL*8 function rcomb()
    INTEGER*4      x/1/,y/3/
    x = MOD(x*1364,1500419)
    y = MOD(y*1528,1400159)
    rcomb = 0.6664804964479922d-6*x + 0.7142046010488808d-6*y
    if (rcomb .ge. 1.0d0) then
        rcomb = rcomb - 1.0d0
    end if
    return
end

cc Sub-rotina rluxin
cc-----
cc  Inicializador do gerador RANLUX (LUXury Random numbers)

```

cc Lüscher (1994), James (1994), Shchur e Butera (1998)

cc Versao: Eloy Kaviski

```

subroutine rluxin(ip,x0,c0)

INTEGER*4      ip,x0,c0

REAL*8         mr/2147483647.0/,y

INTEGER*8      m/2147483647/,k/16807/

INTEGER*2      pp(0:4)/24,48,97,223,389/,l

COMMON         /rluxpa/br,b,x,r,s,p,i,j,n,c

INTEGER*2      r,s,p,i,j,n,c

INTEGER*4      b,x(0:100)

REAL*8         br

b  = 2**24

br = 1.0e0/b

r  = 24

s  = 10

i  = r - 1

j  = r - s - 1

n  = 0

c  = c0

if (ip .ge. r) then

    p = ip

else

    if (ip .lt. 0) ip = 0

    if (ip .gt. 4) ip = 4

    p = pp(ip)

end if

if (x0 .le. 0) then

    x(0) = 524287

else

    x(0) = x0

end if

```

```

do l = 1, r-1
    x(l) = mod(k*x(l-1),m)
end do

do l = 0, r-1
    y      = x(l)/mr
    x(l) = y*b
end do

return

end

cc Funcao ranlux
cc-----
cc Gerador RANLUX (LUXury RANdom numbers)
cc Lüscher (1994), James (1994), Shchur e Butera (1998)
cc Versao: Eloy Kaviski          ranlux(c14)

REAL*8 function ranlux()
COMMON      /rluxpa/br,b,x,r,s,p,i,j,n,c
INTEGER*2   r,s,p,i,j,n,c
INTEGER*4   b,x(0:100)
REAL*8      br

do WHILE (n .le. p)
    if (i .lt. r-1) then
        i = i + 1
    else
        i = 0
    end if
    if (j .lt. r-1) then
        j = j + 1
    else
        j = 0
    end if
    n      = n + 1

```

```

      x(i) = x(j) - x(i) - c
      if (x(i) .lt. 0) then
        x(i) = x(i) + b
        c      = 1
      else
        c = 0
      end if
      if( n .gt. p ) n = 1
      if( n .le. r ) then
        ranlux= x(i)*br
        return
      end if
    end do
  return
end

cc Funcao norbm
cc-----
cc  Geracao de números normais com media 0 e variância 1
cc  Metodo de Box e Müller
cc  Versao: sem calculo de senos e cossenos (Press e outros, 1992) (c15)
      REAL*8 function norbm()
      REAL*8      v1/0.0/,v2/0.0/,f/0.0/,r,ranmul
      logical*1  t/.true./
      if (t) then
10        v1 = 2.0*ranmpb() - 1.0
          v2 = 2.0*ranmpb() - 1.0
          r  = v1*v1 + v2*v2
          if (r .gt. 1.0) go to 10
          f = sqrt(-2.0*log(r)/r)
          norbm = v1*f
          t      = .false.

```

```

        else
            norbm = v2*f
            t      = .true.
        endif
    return
end

cc Funcao poisson
cc-----
cc  Geracao de variaveis aleatorias Poisson com parâmetro lambda
    INTEGER*4 function poisson(lambda)
    REAL*8      lambda,a,c,f
    REAL*4      ranmpb
    INTEGER*4    x
    c = ranmpb()*dexp(lambda)
    x = 0
    a = 1.0d0
    f = a
    do WHILE (f .LT. c)
        x = x + 1
        a = a*lambda/x
        f = f + a
    end do
    poisson = x
    return
end

cc Funcao lngama
cc-----
cc  Logaritmo da Funcao Gama de xx (Press e outros, 1992)
    REAL*4 function lngama(xx)
    REAL*4  xx
    REAL*8  x,tmp,ser,stp

```

```

data      stp/2.50662827465/
x      = xx - 1.0
tmp = x + 5.5
tmp = (x + 0.5)*log(tmp) - tmp
ser = 1.0 + 76.18009173/(x+1.0) - 86.50532033/(x+2.0) +
-          24.01409822/(x+3.0) - 1.231739516/(x+4.0) +
-          0.120858003e-2/(x+5.0) - 0.536382e-5/(x+6.0)

lngama = tmp + log(stp*ser)

return

end

cc -----
cc arquivo datin00.txt
cc
cc  6      1.E-2      3e3      100      1.
cc  n      dt      Tfin      rhosteps alpha
cc  1  1  0  0  0  0  0  0  0
cc  fprint

```